

[19]中华人民共和国专利局

[51]Int.Cl<sup>6</sup>

G06F 13/00



## [12] 发明专利申请公开说明书

[21] 申请号 96121840.1

[43]公开日 1997 年 8 月 6 日

[11] 公开号 CN 1156281A

[22]申请日 96.11.29

[30]优先权

[32]95.11.30[33]JP[31]334179/95

[71]申请人 雅马哈株式会社

地址 日本静冈县

[72]发明人 和智正忠 铃木秀雄

田邑元一 平野正志

[74]专利代理机构 中国国际贸易促进委员会专利商标  
事务所

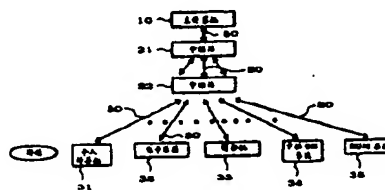
代理人 于 静

权利要求书 7 页 说明书 36 页 附图页数 19 页

[54]发明名称 信息处理系统

[57]摘要

在每个终端设备中, 储存有关于相应设备的用户概貌信息。这信息包括识别用户、设备类型和名称的数据, 关于设备中提供的 CPU、存储器和操作系统的信息, 和关于设备中所储存各种数据和程序的信息。当要从主计算机装入期望数据时, 就把要装入最佳数据的请求送往主计算机。这就容许高效地装入最适合于系统条件的数据。当要从 CD-ROM 之类的记录媒体, 而不从通信网络, 装入数据或程序时, 容许用类似的方式高效地装入之。



(BJ)第 1456 号

## 权 利 要 求 书

---

1.一种信息处理系统包括:

处理装置;

存储装置,连接于所述处理装置,且其中已至少储存了第一数据;

和

数据供给装置,连接于所述处理装置和存储装置,且其中储存了各种数据,

其中,为了根据所述第一数据执行预定的处理,所述处理装置在所述存储装置未储存第二数据时,向所述数据供给装置发送一个请求,以便传送该预定的处理所需的所述第二数据,并且所述供给装置根据传送请求,把所述第二数据传送到所述存储装置.

2.一种音乐信息处理系统包括:

处理装置;

存储装置,连接于所述处理装置,且其中已至少储存了音乐数据;

和

数据供给装置,连接于所述处理装置和存储装置,且其中存储了各种数据,

其中,所述音乐数据包含指定数据,用于对根据所述音乐数据执行预定处理所需的各种数据的音乐相关数据进行指定,并且当所述处理装置根据所述音乐数据执行处理时,所述数据供给装置把由所述指定数据指定的所述音乐相关数据,传送到所述存储装置.

3.一种信息处理系统包括:

一个处理器设备,用于根据第一和第二数据执行预定的处理;

一个存储设备,连接于所述处理设备,且其中至少储存了所述第一数据; 和

一个数据供给设备,通过一个网络连接于所述处理器设备,且其中至少存储了所述第二数据,

其中,所述处理器设备进行一个第一操作,用于确定是否预定处理所需的所述第二数据存储于所述存储设备中; 一个第二操作,用于在所

述第一操作确定，预定处理所需的所述第二数据未储存在所述存储设备时，就请求所述数据供给设备通过网络传送必需的第二数据；和一个第三操作，用于把从所述数据供给设备传送的必需第二数据装入所述存储设备中，以便根据储存在所述存储设备中的所述第一和第二数据，执行预定的处理，和

应所述处理器设备的请求，所述数据供给设备通过网络把必需的数据供给所述处理器设备。

#### 4.一种信息处理系统包括：

一个处理器设备，用于根据第一和第二数据执行预定的处理；

一个存储设备，连接于所述处理器设备，用于在其中储存所述第一和第二数据；和

一个数据供给设备，通过网络连接于所述处理器设备，包括一个在其中储存了多种所述第一和第二数据的数据库，

其中，所述处理器设备进行一个第一操作，用于从多种所述第一数据之中指定期望的第一数据，并且请求所述数据供给设备传送所指定的第一数据；一个第二操作，用于确定是否对应于所指定第一数据使用的所述第二数据，业已储存在所述存储设备中；一个第三操作，用于在所述第二操作确定，所述第二数据未储存在所述存储设备时，请求所述数据供给设备传送所述第二数据；和一个第四操作，用于把从所述数据供给设备传送的所述指定的第一数据和第二数据，装入所述存储设备中，以便根据储存在所述存储设备中的所述第一和第二数据，执行预定的处理，和

应所述处理器设备的请求，所述数据供给设备通过网络，把所述指定的第一数据和所述第二数据供给所述处理器设备。

#### 5.一种信息处理系统包括：

一个处理器设备，用于根据第一和第二数据执行预定的处理；

一个存储设备，连接于所述处理器设备，用于在其中储存所述第一和第二数据；和

一个数据供给设备，通过网络连接于所述处理器设备，包括一个在其中储存了多种所述第一和第二数据的数据库，

其中，所述处理器设备进行一个第一操作，用于从多种所述第一数据之中指定期望的第一数据，并且请求所述数据供给设备传送所指定的第一数据；一个第二操作，用于准许在请求只传送所指定第一数据的一个第一模式，与请求传送所指定第一数据和对应于所指定第一数据要使用的所述第二数据的一个第二模式之间，进行模式的选择；一个第三操作，用于在选择所述第二模式时，确定是否对应于所指定第一数据要使用的所述第二数据，业已储存在所述存储设备中；一个第四操作，用于在所述第三操作确定，所述第二数据未储存在所述存储设备中时，请求所述数据供给设备传送所述第二数据；和一个第五操作，用于把从所述数据供给设备传送的所述指定的第一数据和第二数据，装入所述存储设备中，以便根据储存在所述存储设备中的所述第一和第二数据，执行预定的处理，和

应所述处理器设备的请求，所述数据供给设备通过网络，把所述指定的第一数据和第二数据供给所述处理器设备。

#### 6.一种信息处理系统包括：

一个处理器设备，用于根据第一和第二数据执行预定的处理；

一个存储设备，连接于所述处理器设备，在其中至少存储了所述第一数据；和

一个数据供给设备，连接于所述处理器设备和存储设备，包括一个其中至少储存了所述第二数据的存储器，

其中，所述处理器设备进行一个第一操作，用于确定是否预定处理所需的所述第二数据储存在所述存储设备中；一个第二操作，用于在所述第一操作确定，预定处理所需的所述第二数据未储存在所述存储设备中时，请求所述数据供给设备传送必需的第二数据；和一个第三操作，用于把从所述数据供给设备传送的必需第二数据，装入所述存储设备中，以便根据储存在所述存储设备中的所述第一和第二数据，执行预定的处理，和

应所述处理器设备的请求，所述数据供给设备从所述存储器中读出必需的第二数据，并且把所读出的必需第二数据供给所述处理器设备。

#### 7.一种信息处理系统包括：

一个处理器设备，用于根据第一和第二数据执行预定的处理；

一个存储设备，连接于所述处理器设备，用于在其中储存所述第一和第二数据；和

一个数据供给设备，连接于所述处理器设备和存储设备，包括一个其中存储了多种所述第一和第二数据的存储器；

其中，所述处理器设备进行一个第一操作，用于从多个要执行的预定处理中选择一个处理；一个第二操作，用于从所述数据供给设备的所述存储器中，读出所选择预定处理所需要的所述第一数据，并且把所读出的第一数据装入所述存储设备中；一个第三操作，用于确定是否所选择预定处理所需的所述第二数据，业已储存在所述存储设备中；一个第四操作，用于在所述第三操作确定，所选择预定处理所需的所述第二数据未储存在所述存储设备中时，请求所述数据供给设备传送必需的所述第二数据；和一个第五操作，用于把从所述数据供给设备传送的必需第二数据，装入所述存储设备中，以便根据存储在所述存储设备中的所述第一和第二数据，执行所选择的预定处理，和

应所述处理设备的请求，所述数据供给设备从所述存储器中读出必需的所述第二数据，并且把所读出的必需第二数据供给所述处理设备。

8.根据权利要求 7 所述的信息处理系统，其中，所述第四操作确定是否所述必需的所述第二数据储存在所述数据供给设备的所述存储器中；并且其中，如果所述必需的所述第二数据储存在所述存储器中，则所述第四操作请求所述数据供给设备，传送所述必需的所述第二数据；但如果所述必需的所述第二数据未储存在所述存储器中，则所述第四操作请求所述数据供给设备，传送多种所述第二数据中的任何一种数据，以代替所述必需的所述第二数据。

9.根据权利要求 7 所述的信息处理系统，其中，所述第一数据包括用于指定所述必需第二数据的指定信息；并且其中，所述第三操作借助在所述第二操作中，从所述存储器读出的所述第一数据内包含的指定信息，确定是否所选择预定处理所需的所述第二数据，业已储存在所述存储设备中。

10.一种信息处理系统包括：

一个处理器设备，用于根据包含自动演出音乐数据的第一数据和涉及所述第一数据的第二数据，执行音乐再现处理；

一个存储设备，连接于所述处理器设备，其中至少储存了所述第一数据；和

一个数据供给设备，通过网络连接于所述处理器设备，包括一个其中储存了至少包含多种所述第二数据在内的各种数据的数据库；

其中所述第一数据包括用于指定音乐数据再现所需的所述第二数据的指定信息；

所述处理器设备进行一个第一操作，用于请求所述数据供给设备，通过网络，对由所选择音乐段的所述第一数据中包含的所述指定信息所指定的所述第二数据，进行传送，以便执行一个所选择音乐段的再现演出；和一个第二操作，用于把从所述数据供给设备传送的所述第二数据，装入所述存储设备中，以便根据储存在所述存储设备中的所述第一和第二数据，执行所选择音乐段的音乐再现处理，和

应所述处理设备的请求，所述数据供给设备通过网络，把所述第二数据供给所述处理器设备。

#### 11. 一种信息处理系统包括：

一个处理器设备，用于根据包含自动演出音乐数据的第一数据和涉及所述第一数据的第二数据，执行音乐再现处理；

一个存储设备，连接于所述处理器设备，其中至少储存了所述第一数据；和

一个数据供给设备，连接于所述处理器设备和存储设备，包括一个其中储存了至少包含多种所述第二数据在内的各种数据的存储器；

其中，所述第一数据包括，用于指定音乐数据再现所需的所述第二数据的指定信息；

所述处理器设备进行一个第一操作，用于请求所述数据供给设备，对由所选择音乐段的所述第一数据中包含的所述指定信息所指定的所述第二数据，进行传送，以便执行一个所选择音乐段的再现演出；和一个第二操作，用于把从所述数据供给设备传送的所述第二数据，装入所述存储设备中，以便根据储存在所述存储设备中的所述第一和第二数据，

执行所选择音乐段的音乐再现处理, 和

应所述处理设备的请求, 所述数据供给设备把所述第二数据供给所述处理器设备。

12. 一种把数据从一个系统中的一个数据库传送到一个存储设备的方法, 该系统包括: 该存储设备; 一个处理设备, 用于根据在该存储设备中储存的数据, 执行预定的处理; 和其中储存了各种数据的数据库; 根据专门用于预定处理的第一数据, 和不仅可用于预定处理而且可用于其他处理的第二数据, 执行预定的处理; 所述的方法包括下列诸步骤:

请求所述的数据库, 传送期望的所述第一数据;

确定是否, 根据所述期望第一数据执行预定处理所需的所述第二数据, 储存在所述存储设备中;

在所述确定步骤确定, 所述必需第二数据未储存在所述存储设备中时, 请求所述的数据库, 传送所述的必需第二数据; 和

根据该请求, 把从所述数据库传送的所述第一和第二数据, 装入所述存储设备中。

13. 一种把数据从一个系统中的一个数据库传送到一个存储设备中的方法, 该系统包括: 该存储设备; 一个处理器设备, 用于根据在该存储设备中存储的数据, 执行预定的处理; 和其中存储了各种数据的该数据库; 根据专门用于预定处理的第一数据, 和不仅可用于执行预定处理而且可用于执行其他处理的第二数据, 执行预定的处理; 该存储设备在其中至少储存了所述第一数据; 所述的方法包括下列的诸步骤:

确定是否在要执行预定处理时, 执行预定处理所需的所述第二数据, 储存在所述存储设备中;

在所述确定步骤确定, 所述的必需第二数据未存储于所述存储设备中时, 请求所述的数据库传送所述的必需第二数据; 和

根据该请求, 把从所述数据库传送的所述第二数据, 装入所述存储设备中。

14. 一种机器可读的记录媒体, 包含一组指令, 使所述的机器执行一种把数据从一个系统中的一个数据库传送到一个存储设备中的方法, 该系统包括该存储设备, 一个用于根据储存在该存储设备中的数据来执行

预定处理的处理器设备，和其中储存了各种数据的该数据库；根据专门用于该预定处理的第一数据，和不仅可用于执行该预定处理而且可用于执行其他处理的第二数据，执行该预定处理；所述的方法包括下列的步骤：

请求所述的数据库传送期间的所述第一数据；

确定是否，根据所述期望的第一数据执行预定处理所需的所述第二数据，储存在所述的存储设备中；

在所述确定步骤确定，所述的必需第二数据未存储于所述存储设备中时，请求所述的数据库传送所述的必需第二数据；和

根据该请求，把从所述数据库传送的所述第一和第二数据，装入所述存储设备中。

15.一种机器可读的记录媒体，包含一组指令，使所述的机器执行一种把数据从一个系统中的一个数据库传送到一个存储设备中的方法；该系统包括该存储设备，一个用于根据存储于该存储设备中的数据来执行预定处理的处理器设备，和其中储存了各种数据的数据库；根据专门用于预定处理的第一数据，和不仅可用于执行预定处理而且可用于执行其他处理的第二数据，执行预定的处理；存储设备在其中至少存储了所述的第一数据；所述的方法包括下列的步骤：

确定是否在要执行预定处理时，执行预定处理所需的所述第二数据，存储于所述存储设备中；

在所述确定步骤确定，所述的必需第二数据未存储于所述存储设备中时，请求所述的数据库传送所述的必需第二数据，和

根据该请求，把从所述数据库传送的所述第二数据，装入所述存储设备中。



# 说明书

---

## 信息处理系统

一般说来, 本发明涉及信息处理系统, 更详细地说, 涉及一种经改进的信息处理系统, 它能够把各种来自网络系统、记录媒体或类似媒体的诸如音乐数据之类的数据装入终端设备中。

随着象个人计算机通信网络和互联网 ( Internet ) 之类的计算机网络的普及应用, 已在全球范围内建立了各种使用电话线路网络之类的通信网络来交换各种数据与信息的通信系统。使用上述通信网络可迅速而方便地传输各种数据和信息。用个人计算机处理音乐信息, 也已变得很普及。除了音乐“定序”软件之外, 还开发了卡拉 OK ( 为表演而唱所记录的音乐 ) 软件, 它能用个人计算机实现卡拉 OK 表演。在使用定序软件和卡拉 OK 软件时, 怎样向用户高效率地提供音乐段数据 ( music piece data ) ( 尤其是新版音乐段数据 ) 是一个很重要的问题。

用户能够用通信网络方便而迅速地得到想要的音乐数据。同样, 能够把一个更新版本的象卡拉 OK 或音响发生软件之类的各种软件, 用通信网络交付给用户。

根据已知的方法, 通过用户从主系统的数据目录首选数据, 从网络下载所期望的数据。然而, 在用上述方法时, 和已由用户 ( 终端设备 ) 装入或存入数据相同的数据, 倾向于以覆盖的方式错误地下载, 即, 用户可能选择一些完全无关的数据。此外, 已知的方法需要在用户部分进行麻烦的选择操作, 无意中增加了网络业务量。在利用付费通信网络的场合, 就导致费钱。在把数据从一个 CD - ROM 或其他记录媒体装入一个人计算机的场合, 也能发生同样的问题。

因此, 本发明的一个目的在于提供一种信息处理系统, 它使最佳数据可以通过确定用户系统中的系统和程序处理条件而高效率地装入。

本发明的另一个目的在于提供一种信息处理系统, 它用于把数据或程序管理软件装入用户系统中以及把数据从通信网络采集到用户系统

中，并且容许把预定的数据或程序从一个预定记录媒体有效地装入该系统中。

为了达到上述目的，本发明提供一种信息处理系统，它包括一个处理部分，一个连接于该处理部分且其中至少已储存第一数据的存储部分，和一个连接于该处理部分和存储部分且其中已储存各种数据的数据供给部分；其中，根据第一数据执行预定的处理，处理部分在第二数据未储存在存储部分时数据供给部分发出一个传送第二数据的请求，并且数据供给部分根据传送请求把第二数据传送到存储部分。

在上述信息处理系统的一个典型实例中，把处理部分和存储部分装在一个用户终端中，而把数据供给部分装在一个主设备中，或另一个经通信网络存取的数据库的终端中。另一方面，处理部分可以装在另一个连接于网络的终端或主设备中，并且存储部分可以插入一个终端设备中或连接于网络中。

为了执行预定的处理，除了第一数据以外，还需要第二数据。例如，在预定的处理是自动重放一段给定的音乐（音乐段）的场合，第一数据就包括自动表演序列数据（包括指定音符(note)和乐音(tone)发生定时的数据），这是完全重放该音乐段所需的；而第二数据则包括重放各个乐音所需的波形数据和乐音发生器程序数据。一个终端设备的存储部分已在其中至少存储第一数据，可能还包含某些第二数据。当在终端作出重放一段音乐的选择时，就根据第一数据执行预定的处理（在这种情况下，执行自动音乐段重放处理），但对完整地执行预定的处理来说，则需要存储部分拥有的或储存的第二数据。处理部分确定预定处理所需的第二数据是否储存在存储部分中。如果预定处理所需的全部或一些第二数据未存于存储部分，则处理部分向数据供给部分发出一个传送所缺数据的请求。如果数据供给部分是以整个信息处理系统的数据库形式出现，则该数据库含有该系统所需的全部数据（或者至少全部第二数据），从而能够根据传送请求把所请求的第二数据传送到存储部分。该处理部分，或者数据供给部分的处理器，本身可以根据终端的传送请求，确定是否第二数据存于存储部分中。

这种布置容许选择地只传送那些实际上未存于存储部分的第二数

据，从而实现高效率地下载。

在如此设计本发明的情况下，能够根据用户终端的系统和程序处理条件，自动下载最佳数据或程序。此外，能够在没有用户麻烦选择操作情况下，方便地选择要装入的所需数据，并且能够防止把那些与已由用户终端处理的数据相同的数据，以重叠的方式装入用户终端，从而能够避免空处理。于是，本发明具有较多的好处，例如可有效地防止增加网络业务量。

在信息处理系统的另一实例中，数据供给部分是不用介入通信网络而连接于处理部分和存储部分的。在这种情况下，数据供给部分也可装在一个用户终端中，并且本发明的原则可用于把数据或程序的管理软件装入用户系统内。数据供给部分是以一种具有较小容量的数据库形式出现的，例如一个 CD - ROM 之类的可携带的记录媒体。

这样，甚至当把数据或程序从一个 CD - ROM 之类的记录媒体装入一个用户系统内的象 RAM 之类的内部存储器时，也能够在免除用户麻烦选择操作的情况下方便地选择要装入的所需数据，并且能够防止把和已存入内部存储器的数据相同的数据，以重叠的方式装入，从而能够避免空处理。

本发明的另一个方面提供一种音乐信息处理系统，它包括一个处理部分，一个连接于该处理部分且其中已至少存有音乐数据的存储部分，和一个连接于该处理部分和存储部分且其中已存有各种数据的数据供给部分；其中，音乐数据包含用于标志各种数据中的音乐相关数据的标志数据，这些数据是根据音乐数据执行预定的处理所需的，并且当处理部分根据音乐数据执行处理时，数据供给部分把用标志数据标志的音乐相关数据传送到存储部分。

此外，在这种情况下，如果预定的处理是自动重放一段给定的音乐，则音乐数据包括自动表演序列数据（包含指定音符和乐音发生时序的数据），它们是全部重放该音乐段所需的，并且音乐相关数据包括重放各个乐音所需的波形数据和乐音发生器程序数据。存储部分已在其中至少存有音乐数据，并且可能还包括一些音乐相关数据。音乐信息处理系统的特点在于，音乐数据不但包括自动表演序列数据，而且包括对执行预

定的处理（自动音乐段重放处理）所需的音乐相关数据进行标志的那些数据。只有那些重放一段音乐所需的音乐相关数据（波形数据或乐音发生器程序数据），才可以根据标志数据从数据供给部分选择性地转移，然后存入存储部分中。应当按照所需的音乐相关数据供给何种数据，是由标志数据自动地规定的，这能减轻用户在装入数据方面的负担。此外，因为标志数据的数量远少于音乐相关数据的数量，故能显著地减少预先存于存储部分的音乐数据的总量。

为了更好地了解本发明，下面参照附图，详述本发明的一些优选实施例，其中：

图 1 是一个方块图，说明根据本发明的信息处理系统的实施例的一般结构；

图 2A 是一个方块图，说明图 1 中主计算机的典型结构；

图 2B 说明图 2 的数据库中储存的一些数据组的实例；

图 3A 是一个方块图，说明图 1 中终端设备的一个典型结构；

图 3B 是图 1 的个人计算机的 RAM 中典型存储变换的说明图；

图 4A 是音乐段数据的典型结构的说明图；

图 4B 是波形数据的典型结构的说明图；

图 5A 是参数数据的典型结构的说明图；

图 5B 是程序数据的典型结构的说明图；

图 5C 是用户简要信息的典型数据结构的说明图；

图 6 是一个流程图，说明由图 1 的主计算机执行的主程序的实例；

图 7 是一个流程图，说明数据传送过程实例的一部分；

图 8 是一个流程图，说明图 7 的数据传送过程的另一部分；

图 9 是一个流程图，说明图 7 的数据传送过程的剩余部分；

图 10 是一个流程图，说明由图 1 的终端设备执行的主程序的实例；

图 11 是一个流程图，说明网络处理实例的一部分；

图 12 是一个流程图，说明图 11 的网络处理的另一部分；

图 13 是一个流程图，说明图 11 的网络处理的又一个部分；

图 14 是一个流程图，说明图 11 的网络处理的剩余部分；

图 15 是一个流程图，说明一个应用程序处理实例；

图 16 是一个流程图，说明一个音乐段数据表演处理的实例；

图 17 是一个流程图，说明一个音乐段数据选择处理的实例；

图 18 是一个流程图，说明一个乐音发生处理的实例；

图 19 是一个流程图，说明一个波形计算的实例；和

图 20 是一个时序图，解释图 18 的乐音发生处理。

下面就一种用来供给（交付或装入）各种音乐数据，例如音乐段数据和波形数据以及乐音发生程序的信息处理系统，描述本发明的一些优选实施例。

#### （系统的一般结构）

图 1 是一个方块图，说明一种根据本发明的信息处理系统的实施例的一般结构。如图所示，信息处理系统一般包括一个主计算机 10，中继站 21 和 22，诸如公用电话线路之类的网络线路 50，通过网络线路 50 和中继站 21，22 而连接于主计算机 10 的终端设备 31 至 35。终端设备 31 至 35 包括一个人计算机 31，一个电子乐器（E.M.I.）32，一个博弈机 33，一个通信卡拉 OK 系统 34，和一个例如以电缆广播为基础的 BGM 系统。终端设备 31 至 35 中的每一个，都在相应线路 50 是一个公用电话线路时用一个 MODEM（调制解调器）连接于网络线路 50，并且在相应的网络线路 50 是一个专用的数字线路时，还具有按照预定的协议以预定的格式来发送和接收数据的功能。除了上述终端设备以外的任何类型的终端设备，只要它们可连接于网络，都可以使用。

中继站 21 和 22 在所用线路是商用电路线路时可以是本地电话局，或者在所用线路是 LAN（局域网）线路时可以包括一些计算机。如果该网络是小规模网络，则可去掉中继站 21 和 22，使主计算机 10 直接连接于终端设备 31 至 35。通信网络线路可以部分地依靠通过微波的地面电波通信，或者通过卫星的无线电通信。此外，作为数据通信方法，除了通常已知的数字通信方法之外，还可以采用作为话音通信基础的传送数字数据的方法。

为了响应从终端设备 31 至 35 中任何一个设备发出的下载请求，或主计算机 10 上作出的标志，计算机 10 就把期望的音乐段数据、用于表演音乐段的表演软件，或用于模拟乐音发生器（软件乐音发生器）的

乐音发生器 ( T.G. ) 软件, 发送到终端设备。

图 2A 示出主计算机 10 的一种典型结构。如图 2A 所示, 主计算机 10 包括一个中央处理单元 ( CPU ) 11, 一个已在其中预先储存各种控制程序 and 数据的存储器 12, 一个已在其中预先储存音乐段数据、波形数据和各种表演数据以及乐音发生器软件的数据库 13, 一个控制台和显示器 14, 一个连接于网络线路 50 的诸如 MODEM 之类的网络接口电路 15, 和一个总线 16。

图 2B 示出存于数据库 13 的多个数据组的实例: 音乐段数据组 ( MUSIC DATA ( FILE ) 1 - K ) 131; 波形数据组 ( WAVE DATA ( FILE ) 1 - L ) 132; 乐音生成参数组 ( PARA DATA ( FILE ) 1 - M ) 133; 乐音发生程序组 ( TONE PGM ( FILE ) 1 - N ) 134, 例如各种用于乐音发生器的控制程序; 和表演处理程序组 ( PLAY PRG ( FILE ) 1 - P ) 135, 例如自动表演程序和卡拉 OK 软件。这些数据组可以存储于任何适合的存储或记录媒体中, 例如软磁盘 ( FD )、硬磁盘 ( HD )、磁光盘 ( MO )、或 CD - ROM 中。在这个实施例中, 音乐段数据组 ( MUSIC DATA ( FILE ) ) 包括用于实现自动表演的数据。

可连接于网络的终端设备 31 至 35 中的每个设备全是计算机系统, 虽然它们在外特征、操作/显示部分、存储软件等方面, 随着各种设备 ( 即个人计算机 ( PC )、卡拉 OK 设备、博弈机等 ) 的不同而互相不同, 但它们基本上具有如图 3A 所示的类似结构。如图 3A 所示, 终端设备 31 至 35 中的每个设备都包括一个 CPU 101, 用于控制终端设备的全部操作; 一个 ROM 102, 已在其中预先存储控制程序等; 一个 RAM 103, 用于存储各种数据; 一个硬盘设备 104; 一个软盘设备 105; 一个 CD - ROM 或 MO 驱动器 106; 一个插件接口电路 107; 一个存储插件 108; 一个象 MODEM 之类的网络接口电路 109; 一个象字符键盘之类的控制台 110; 一个显示设备 111; 一个在终端设备是一件电子乐器场合使用的音乐键盘 112; 一个乐音发生器 ( T.G. ) 113; 和一个系统信号通路 114。在这个实施例中, 从种属上说, “信号通路” 114 不仅涉及一个 CPU 总线, 而且涉及一个扩展总线, SCSI ( 小型计算机系统接口 ) 和根

据其他连接标准的其他连接。

在这个实施例中，按照终端设备的技术要求，把包括硬盘设备 104、软盘设备 105、CD - ROM 或 MO 驱动器 106 在内的各种存储或记录媒体，连接到终端设备上；媒体装配和连接用的技术要求是随着终端设备类型的不同而不同的。在终端设备要连接于 CPU 总线的场合，可以用各种方式，例如通过扩展接口电路或 SCSI 之类的接口，来连接存储媒体。

可以实现乐音发生器 113 的方法有：只通过一个包括一个数模转换器（D/A）的 CODEC（编码译码器），或者通过把一个 CODEC 同包括专用 LSI、DSP 或 MPU 在内的乐音发生器硬件结合起来，或者借助一个 MIDI（乐器数字接口）来驱动一个乐音发生器设备。在只用一个 CODEC 实现乐音发生器 113 的场合，CODEC 同由软件按算术运算方式生成乐音波形数据的所谓软件乐音发生器一起使用。通过把 CODEC 同乐音发生器硬件相结合的方法来实现乐音发生器 113 的场合，可以把一个装有一个乐音发生器芯片或设备的插件板（子插件板），单独地装到 CODEC 上；或者可以把这些元件从一开始就共同装到 CODEC 上，即，可以把这些元件和 CODEC 都并入一单个 LSI 中。

图 3B 示出个人计算机 31 的 RAM 103 中一种典型的存储变换，在此只有 CODEC 被安装，用作由软件生成乐音波形数据的乐音发生器 11。如图所示，在存储区 1031 存储一个人计算机 31 的操作系统，在存储区 1032 存储多种表演处理程序，且在存储区 1033 中储存一个或多个乐音发生程序。此外，在存储区 1034 储存一组乐音波形数据（WAVE DATA），且在存储区 1035 存储要播放的音乐段数据（MUSIC DATA）。此外，存储区 1036 在其中存储各种其他的数据和程序，或者有时被置于一种未占用状态。

#### （数据结构）

下面描述本发明所用的各种数据结构。

图 4A 示出音乐段数据集或文件（MUSIC DATA（FILE））的典型结构，它一般包括一个主音乐段数据部分（MUSIC DATA）和一个伴随该主部分的音乐段相关数据部分（MUSIC MISC DATA）。主音乐

段数据部分 ( MUSIC DATA ) 是以压缩格式储存于数据库 13 的, 而音乐段相关数据部分 ( MUSIC MISC DATA ) 则以非压缩格式储存于数据库 13 中。主音乐段数据部分 ( MUSIC DATA ) 包括: 识别音乐段名称的数据 ( SONG NAME ), 识别音乐段数据的版本号的数据 ( VERN O ID ); 在音乐段数据集针对卡拉 OK 唱歌的场合, 对用于直观地显示音乐段字词的语言进行识别的数据 ( LANGUAGE ID ); 指示音乐段表演速率的数据 ( TEMPO ); 指示音乐段节拍的数据 ( BEAT ); 主表演事件数据, 其中包含那些识别音乐段中各个表演事件的类型与发生时间的数据; 和指示主音乐段数据部分结束的数据 ( END OF DATA )。主表演事件数据部分还包含关于乐音发生器和要用的乐音发生程序的信息 ( TG INFO )。

在音乐段数据用于卡拉 OK 唱歌的场合, 主表演事件数据中包括字词和图象数据。换句话说, 可以分别存储表演事件数据和字词 (与图象) 数据, 以便根据音乐段应用于卡拉 OK 唱歌而不用于仅仅是演出的选择, 读出字词 (与图象) 数据, 从而执行有直观显示的字词或图象的演出。

音乐段相关数据部分 ( MUSIC MISC DATA ) 是伴随主音乐段数据部分的, 它包括: 识别音乐段名称的数据 ( SONG NAME ); 包含音乐段简介的信息 ( MUSIC INFO ) 和类似信息; 识别音乐段数据的版本号的数据 ( SONG VERN O ID ); 关于乐音发生器和涉及音乐段数据的乐音发生程序的信息 ( TG INFO ); 和指示音乐段数据结束的数据 ( END OF FILE )。根据情况, 音乐段相关数据部分 ( MUSIC MISC DATA ) 也可包括一个音色表, 其中列举要在音乐段表演中使用的全部音色的号或名。

图 4B 示出波形数据集或文件 ( WAVE DATA ( FILE ) ) 的典型结构, 它用于生成波形的乐音发生器, 通常包括主波形数据部分 ( WAVE DATA ) 和伴随该主波形数据部分的波形相关数据部分 ( WAVE MISC DATA )。主波形数据部分 ( WAVE DATA ) 以压缩格式存储于数据库 13 中, 而波形相关数据部分 ( MUSIC MISC DATA ) 则以非压缩格式存储于数据库 13 中。主波形数据部分 ( WAVE DATA ) 包括: 识别波形名称的数据 ( WAVE NAME ); 识别波形数据的版本号的数据



( WAVE VER NO ID ) ; 识别波形数据的格式的信息 ( WAVE FORMAT ID ) ; 波形样本数据 ( WAVE SAMPLE DATA ) ; 和指示波形数据结束的数据 ( END OF DATA ) .

波形相关数据部分 ( WAVE MISC DATA ) 包括: 识别波形名称的数据 ( WAVE NAME ); 识别波形数据的版本号的数据 ( WAVE VER ID ); 包含波形数据简介的信息 ( WAVE MISC INFO ) 和类似信息; 和指示波形数据结束的数据 ( END OF DATA ) .

在图 5A 中, 示出乐音生成参数数据集或文件 ( PARAMETER DATA ( FILE ) ) 的典型结构, 由一个旨在运算地生成一个波形而不用波形数据的乐音发生器, 来使用它, 它通常包括主参数数据部分 ( PARAMETER DATA ), 和伴随该主参数数据部分的参数相关数据部分 ( PARAM MISC DATA ). 主参数数据部分 ( PARAMETER DATA ) 以压缩格式存储于数据库 13 中, 而参数相关数据部分 ( PARAM MISC DATA ) 则以非压缩格式存储于数据库 13 中. 主参数数据部分 ( PARAMETER DATA ) 包括: 识别乐音生成参数数据的名称的数据 ( PARAM NAME ); 识别乐音生成参数数据的版本号的数据 ( PARAM VER ID ); 识别乐音生成参数数据的种类的数据 ( PARAM KIND ID ); 参数数据 ( PARAMETER ); 和指示乐音生成参数数据的结束的数据 ( PARAM VER ID ) .

参数相关数据部分 ( PARAM MISC DATA ) 包括: 识别乐音生成参数数据的名称的数据 ( PARAM NAME ); 识别乐音生成参数数据的版本号的数据 ( PARAM VER ID ); 包含乐音生成参数数据的简介的信息 ( PARAM MISC INFO ) 和类似的信息; 和指示文件结束的数据 ( END OF FILE ) .

乐音生成程序数据集 ( 文件 ) ( TONE PGM DATA ( FILE ) ) 和表演处理程序数据集 ( 文件 ) ( PLAY PGM DATA ( FILE ) ), 都具有如图 5B 所示的相同的数据结构. 这就是说, 这些程序数据集中的每一个都包括主程序数据部分 ( PROGRAM DATA ) 和程序相关数据部分 ( PROGRAM MISC DATA ). 主程序数据部分 ( PROGRAM DATA ) 以压缩格式存储于数据库 13 中, 而程序相关数据部分 ( PROGRAM

MISC DATA ) 则以非压缩格式存储于数据库 13 中。如图所示, 程序相关数据部分 ( PROGRAM MISC DATA ) 包括: 识别乐音发生程序或表演处理程序的名称的数据 ( PROGRAM NAME ); 识别程序的版本号的数据 ( PRGM VER ID ); 包含程序的简介的信息 ( PRGM MISC INFO ); 和指示文件结束的数据 ( END OF FILE ) 。

因为每个上述数据集的主数据部分均以压缩格式而存储, 并且这些数据集中的每一个数据集的相关数据部分均以非压缩格式而存储, 故能够在相应的主数据部分下载之前, 参考相关数据部分, 这会对主数据部分的下载是否有效一事, 有效地促进管理。当用户装入压缩数据集时, 所装入的压缩数据集将由已接收该数据集的终端设备恢复到原有的格式。在一种执行中, 可以把某些数据附加到任何一个上述的数据集中, 以规定该数据集的使用期限, 使相应的程序能够根据所规定的使用期限, 确定该数据集是否能够在某一特定时间使用。在这种情况下, 能够防止该程序数据集在使用期限已过之后被启用。

在终端设备 31 至 35 中的每个设备中, 都存储关于终端设备本身的信息, 即, 用户概貌信息 ( USER PROFILE )。在上述用户概貌信息也存储于主计算机 10 的场合, 能够根据用户概貌信息 ( USER PROFILE ) 执行各种处理。图 5C 示出用户概貌信息 ( USER PROFILE ) 的典型结构, 它包括用户标识信息 ( USER ID ), 用户个人信息 ( USER PERSONAL ID ), 用户系统信息 ( USER SYSTEM INFO ), 和目录信息 ( DIRECTORY INFO )。

用户个人信息 ( USER PERSONAL ) 包括用户名称 ( USER NAME ) 和用户地址 ( USER ADDRESS )。用户系统信息 ( USER SYSTEM INFO ) 包括: 指示设备种类的类型数据 ( MACHINE KIND ), 例如卡拉 OK 设备, 个人计算机或博弈机; 用其型号识别设备名称的名称数据 ( MACHINE NAME ); 关于设备所用 CPU 的信息 ( CPU INFO ); 设备中提供的存储容量之类的存储信息 ( MEMORY INFO ); 关于设备中所提供操作系统的信息, 例如它的版本和类型 ( OS INFO ); 和关于设备中所用网络协议的信息 ( PROTOCOL INFO )。

目录信息 ( DIRECTORY INFO ) 包括一个预先储存在设备中的波

形数据表 ( WAVE LIST ) , 和一个乐音发生程序或表演处理程序的表 ( SOFT LIST ) .

( 主计算机 10 中的处理 )

图 6 是一个流程图, 说明由主计算机 10 执行的主程序的实例. 主计算机 10 通过分时处理 ( TSS ) 为最大数量的网络通道 MAX NET CH 进行线路服务, 以便根据任何一个终端设备发布的下装请求来传送数据.

首先, 在图 6 的步骤 S10, 把指示一个要服务的网络线路的指示字  $i$  初始化成 “ 1 ” 值. 然后, 主计算机 10 转到步骤 S20, 在此为指示字  $i$  指示的线路 ( 即, 第 “  $i$  ” 个线路 ) 执行数据传送处理. 此后, 在步骤 S30, 使指示字值加 1 (  $i+1$  ), 并在步骤 S40 作一确定: 指示字  $i$  的当前值是否已超过连接于主计算机 10 的通道的最大数目值 ( 即,  $i=\text{MAX NET CH}+1$  ). 如果在步骤 S40 回答否 ( NO ), 则主计算机 10 返回到步骤 S20, 以便对下一个通道 (  $i+1$  ) 进行数据传送处理. 如果指示字  $i$  的当前值按照在步骤 S40 的确定, 已超过通道的最大数目值, 则主计算机 10 回到步骤 S10, 在此又把指示字  $i$  设置到 “ 1 ”, 重复从第一通道开始的数据传送处理.

图 7 至 9 都是流程图, 说明对由指示字  $i$  指示的一个网络线路进行的数据传送处理的细节. 在这一数据传送处理中, 通过使用 7 个标志 SEQFLGi1 至 SEQFLGi7 检测第  $i$  个线路的当前状态, 来设置处理的顺序. 全部标志 SEQFLGi1 至 SEQFLGi7 都在信息处理系统启动时, 被初始化成 “ 0 ” 值, 然后在连接第  $i$  个线路时使标志 SEQFLGi1 保持在 “ 1 ”. 并且在完成一个 LOGIN 处理时使标志 SEQFLGi2 设置成 “ 1 ”. 此后, 在从任何一个终端设备输入一个数据下载命令时, 标志 SEQFLGi3 设置成 “ 1 ”, 并且使用标志 SEQFLGi4 至 SEQFLGi7 中的一个来指示一种由输入的数据装入命令所请求的数据类型.

这就是说, 在输入一个音乐段数据下载命令时, 使标志 SEQFLGi4 指示 “ 1 ”; 在输入一个波形数据下载命令时, 使标志 SEQFLGi5 指示 “ 1 ”; 在输入一个参数数据下载命令时, 使标志 SEQFLGi6 指示 “ 1 ”; 并在输入一个程序数据下载命令时, 使标志 SEQFLGi7 指示 “ 1 ”. 此外, 在完成响应于下载命令的数据传送处理时, 使标志 SEQFLGi3 和一

个在标志 SEQFLGi4 至 SEQFLGi7 中相当于所传送数据的标志重新设置成“0”。在图 7 至 9 和下列描述中，为简化起见，在每个上述标志的参考字符结束处所写的字母“i”均被略去。

在图 7 中，在步骤 S20 为第“i”个线路启动该数据传送处理以后，就在步骤 S201 首先作出确定：标志 SEQFLG1 是否处于“0”。如果在该系统处于初始阶段时，例如第“i”个线路未连接，则标志 SEQFLG1 处于“0”，并且因此在步骤 S201 得到一个肯定的确定结果，从而主计算机转到步骤 S202。在步骤 S202，主计算机 10 检查第“i”个线路的当前连接状态。如果在步骤 S203 确定：未连接第“i”个线路，则主计算机 10 回到图 6 主程序的步骤 S30，以进行下一个线路的操作。如果第“i”个线路被连接，并因此而步骤 S203 得到肯定的（YES）确定结果，则主计算机 10 转到步骤 S204，使“1”置入标志 SEQFLG1，从而终止对第“i”个线路的处理，然后回到图 6 的主程序。

当在下一个循环的步骤 S20 为第“i”个线路启动数据传送处理时，就在步骤 S201 得到一个否定的（NO）确定，因为标志 SEQFLG1 在上一次执行步骤 S204 时被设置成“1”。因此，主计算机 10 在步骤 S205 检查第“i”个线路的当前连接状态。如果在步骤 S206 确定：第“i”个线路被连接，则主计算机 10 转到步骤 S207。当在步骤 S206 由于用户的 LOGOUT 命令或系统发出的强制断开命令而使该确定变成否定时，就在步骤 S211 执行一个线路断开处理，在步骤 S212 使全部标志 SEQFLG1 至 SEQFLG7 复位到“0”，并且终止第“i”个线路的数据传送处理。

当该判定在步骤 S206 变成肯定时，主计算机 10 就转到步骤 S207，在此进一步确定：标志 SEQFLG2 是否为“0”。因为现在只有标志 SEQFLG1 由于上一次执行步骤 S204 而设置成“1”，且标志 SEQFLG2 仍然处于“0”，故在步骤 S207 得到一个肯定的确定，从而主计算机 10 移到步骤 S208，以进行一个 LOGIN 处理。在步骤 S208 的 LOGIN 处理中，主计算机 10 访问用户网络，以检查用户 ID，口令等。在下一个步骤 S209，作出判断：是否已完成 LOGIN 处理。如果已完成 LOGIN 处理，就在步骤 S210 使标志 SEQFLG2 设置成“1”，并且对当前由指示字 i 指示的线路，终止步骤 S20 的数据传送处理。如果在步骤 S209 确定：

尚未完成 LOGIN 处理，就终止步骤 S20 的数据传送处理，而不进行步骤 S210 的操作。

当已完成 LOGIN 处理时，如上所述，使 SEQFLG2 设置成“1”，并因此在步骤 S207，使该判断结果变成否定，从而主计算机 10 转到图 8 的步骤 S220。在步骤 S220，就标志 SEQFLG3 是否设置成“0”一事作出确定。因为这时标志 SEQFLG3 处于“0”，故该确定变成肯定，并且主计算机 10 因此而转到步骤 S221，以执行一个终端命令接收处理。在下一个步骤 S222，就从用户终端输入的命令的接收是否已完成一事作出确定。当已完成从用户终端输入的命令的接收，并且在步骤 S222 该确定为否定时，就终止当前线路的处理，并且主计算机 10 回到图 6 的主程序。当步骤 S222 确定，已完成从用户终端输入的命令的接收时，主计算机 1 转到步骤 S223。在步骤 S223 及其后的操作目的在于，使任何一个标志去起动或触发一种相应于所接收用户命令的功能。

在步骤 S223，确定从终端设备输入的命令是否是一个音乐段数据下载命令。在判断结果为肯定是，主计算机 10 转到步骤 S224，在此使标志 SEQFLG3 和 SEQFLG4 设置成“1”，以终止对第“i”个线路的处理。如果相反，从终端设备输入的命令不是一个音乐段数据装入命令，则主计算机 10 分支到步骤 S225，以进一步确定是否从终端设备输入的命令是一个波形数据下载命令。如果在步骤 S225 给出肯定的回答，则主计算机 10 转到步骤 S226，在此使标志 SEQFLG3 和 SEQFLG5 设置成“1”，以终止对第“i”个线路的处理。

如果在步骤 S225 给出否定的回答，则主计算机 10 分支到步骤 S227，以进一步确定是否从终端设备输入的命令是一个参数数据下载命令。在肯定的确定情况下，主计算机 10 转到步骤 S228，在此使标志 SEQFLG3 和 SEQFLG6 设置成“1”，并且终止对第“i”个线路的处理。如果相反，从终端设备输入的命令不是一个参数数据下载命令，则主计算机 10 分支到步骤 S229，以进一步确定是否从终端设备输入的命令是一个程序数据下载命令。如果在步骤 S229 的回答是肯定的，则主计算机 10 转到步骤 S230，在此使标志 SEQFLG3 和 SEQFLG7 设置成“1”，并且终止对第“i”个线路的处理。如果在步骤 S229 的应答是

否定的，就终止对第“i”个线路的处理，而不进行步骤 S229 的操作。

如上所述，当从终端设备输入的命令是一个音乐段数据下载命令时，就使标志 SEQFLG3 和 SEQFLG4 设置成“1”；当输入的命令是一个波形数据下装入命令时，就使标志 SEQFLG3 和 SEQFLG5 设置成“1”；当输入命令是一个参数数据下载命令时，就使标志 SEQFLG3 和 SEQFLG6 设置成“1”；和当输入命令是一个程序数据下载命令时，就使标志 SEQFLG3 和 SEQFLG7 设置成“1”。也就是说，每当从一个终端设备输入任何命令时，标志 SEQFLG3 总是设置成“1”，并且用标志 SEQFLG4 至 SEQFLG7 中的一个标志来指定输入命令的种类。

因为在从任何一个终端设备输入的命令的接收被完成以后，使标志 SEQFLG3 设置成“1”，故在步骤 S220 的判断结果变成否定的，从而主计算机 10 转到图 9 的步骤 S240，以便在步骤 S240 和在其后，执行相应于从终端设备输入的命令的功能。在步骤 S240，就 SEQFLG4 是否处于“1”一事作出确定。如果在步骤 S240 的回答是肯定的，就意味着，从终端设备输入的命令是一个音乐段数据装入命令，于是主计算机转到步骤 S241，在此从数据库 13 读出由命令指定的音乐段数据，并把这些数据传送到连接于第“i”个线路的终端设备。其后，在步骤 S242 确定：是否已完成所指定音乐段数据的传输。如果尚未完成该传输，则主计算机 10 回到主程序；而如果已完成该传输，就在步骤 S243 使标志 SEQFLG3 和 SEQFLG4 都复位成“0”，并且终止当前执行的处理。

如果在步骤 S240 的回答是否定的，则主计算机 10 分支到步骤 S244，在此进一步确定：是否 SEQFLG5 处于“1”。如果在步骤 S244 的回答是肯定的，就意味着，从终端设备输入的命令是一个波形数据下载命令，并且主计算机 10 因此而转到步骤 S245，在此从数据库 13 读出由该命令指定的波形数据，并且把它们传送到该终端设备。其后，在步骤 S246 确定：是否已完成所指定波形数据的传输。如果经步骤 S246 确定：尚未完成该传输，则主计算机 10 回到主程序；而如果已完成该传输，就在步骤 S247 使标志 SEQFLG3 和 SEQFLG5 都复位成“0”，以终止当前执行的处理。

如果在步骤 S244 的应答是否定的，则计算机 10 分支到步骤 S248，

在此进一步确定，是否 SEQFLG6 处于“1”。如果在步骤 S248 的应答是肯定的，就意味着，从终端设备输入的命令是一个参数数据下载命令，于是主计算机 10 转到步骤 S249，在此从数据库 13 读出由该命令指定的参数数据，并且把它们传送到该终端设备。其后，在步骤 S250 确定，是否已完成所指定参数数据的传输。如果经步骤 S250 确定，尚未完成该传输，则主计算机 10 回到主程序；而如果已完成该传输，就在步骤 S251 使标志 SEQFLG3 和 SEQFLG6 都复位成“0”，以终止当前执行的处理。

最后，如果在步骤 S248 的回答是否定的，则主计算机 10 分支到步骤 S252，在此进一步确定，是否 SEQFLG7 处于“1”。如果在步骤 S252 的回答是肯定的，就意味着，从终端设备输入的命令是一个乐音发生或表演处理程序下载命令，于是主计算机 10 转到步骤 S253，在此从数据库 13 读出由该命令指定的程序，并把它传送到该终端设备。其后，在步骤 S254 确定，是否已完成所指定程序的传输。如果尚未完成该传输，则主计算机 10 回到主程序；而如果已完成该传输，就在步骤 S255 使标志 SEQFLG3 和 SEQFLG7 都复位成“0”，以终止当前执行的处理。如果经步骤 S252 确定，SEQFLG7 不处于“1”，也终止当前执行的处理。

按照上述方式，根据从终端设备输入的下载命令的种类，从数据库 13 读出相应的数据，以便把它们传输到该终端设备。在已通过步骤 S240 至 S255 的操作而完成数据传输的时候，使标志 SEQFLG1 和 SEQFLG2 设置成“1”，并且通过从任何一个终端设备接收一个下载命令，能够相继地下装下一个数据。

#### （终端设备中的处理）

下面的段落描述在终端设备中执行的各种处理。图 10 是一个由终端设备 31 至 35 中每个设备执行的主例程序的流程图。如图所示，先在步骤 S50 执行一种初始化处理，它包括把系统程序装入终端设备。其后，在步骤 S60 执行一个系统管理处理，并且在下一个步骤 S70 执行一个应用程序处理。

在此所用的术语“应用程序”不仅涉及在个人计算机或类似设备上

运行的一种普通应用程序，而且涉及一种用于进行一些针对终端设备的预定操作的程序（例如，如果终端设备是一个博弈机，则应用程序是博弈软件）。根据系统的性质，可以通过协同操作系统工作的驱动器软件，执行预定的操作，并且在“应用程序”的范畴内包括这类驱动器软件。

在步骤 S70 的应用程序处理以后，在步骤 S80 执行网络处理，然后程序返回到步骤 S60，以重复进行步骤 S60 的系统管理处理，步骤 S70 的应用程序处理，和步骤 S80 的网络处理。

（网络处理）

下面参照图 11 至 14 详述步骤 S80 的网络处理。网络处理用来把终端设备连接于主计算机 10 上，和下载音乐段数据、波形数据、参数数据、乐音发生器程序和表演处理程序之类的的数据。对于音乐段数据的下载，提供两种方式，其中一种方式是乐音发生器程序伴随方式，用此方式把音乐段数据与相应的乐音发生器程序一起下载；另一种方式是波形数据伴随方式，用此方式把音乐段数据与相应的波形数据一起下装。

使用上述的乐音发生器程序伴随方式和波形数据伴随方式，容许一个用户自动地下载一个最佳的乐音发生器程序或波形数据。按照乐音发生器程序伴随方式，能够把一个乐音发生器程序以及音乐段数据供给一个没有连接于网络 50 的共鸣板或外部乐音发生器的用户，从而能够很方便地演出一个音乐段。另一方面，按照波形数据伴随方式，能够把获得一个音乐段的独特音乐表达所需的特殊波形数据，供给一个只有标准波形数据的用户，从而能够方便地实现高质量的演出。

在步骤 S80 的网络处理中，通过使用在初始阶段都复位成“0”的标志 TERM1 - TERM3 和 TERM11 - TERM22，控制操作顺序。

在启动网络处理时，在步骤 S801 进行一种命令/设置操作处理，以确定由用户操作的鼠标器、键盘或操作面板的当前操作状态。在下一个步骤 S802 执行一种主响应提示接收与显示处理，然后在步骤 S803 确定：是否标志 TERM1 当前处于“0”值。因为标志 TERM1 在初始阶段处于“0”，故在步骤 S803 得出肯定的判断结果，使处理推进到步骤 S804，以执行一个访问点连接和 LOGIN 处理。

在这种情况下，可以对一个最近的访问点或对一个需要最低线路成



本的访问点，进行自动存取。在下载数据时，可以用一个当前连接的访问点作数据选择标准。例如，在用长途呼叫进行存取の場合，可以自动地选择一个具有最低数据量的相同音乐段或程序的版本，供下载之用。

在步骤 S804 之后，处理移到步骤 S805，以确定是否已完成 LOGIN 处理。如果已完成 LOGIN 处理，则在步骤 S806 把标志 TERM1 设置成“1”以后，该处理终止网络处理。

当在步骤 S803 由于标志 TERM1 被设置成“1”而使该判断结果变成否定以后，处理就分支到步骤 S807，以检查当前的线路连接状态和 LOGOUT 操作，从而在步骤 S808 确定：线路是否被连接。如果在步骤 S808 以肯定来应答，则处理推进到步骤 S809，以便进一步确定，是否标志 TERM2 当前被设置成“0”。如果在步骤 S810 回答为否定，则处理进到步骤 S810，以进一步确定，是否标志 TERM3 当前被设置成“0”。这样，通过步骤 S809 和 S810，根据标志 TERM2 和 TERM3 的当前值，进行不同的相继操作。

当在步骤 S808 使该判断结果变成否定以后，处理就分支到步骤 S811，以执行一个线路断开/LOGOUT 对应处理。其后，该处理在步骤 S812 使全部 TERM 标志复位成“0”以后，终止网络处理。

在已完成对主计算机 10 的 LOGIN 的时候，标志 TERM2 仍然处于“0”，就在步骤 S809 使该判定结果变成肯定，并且处理移到步骤 S820（图 12），在此确定是否标志 TERM11 在当前处于“0”。因为标志 TERM11 在这时处于“0”，故该判断结果在步骤 S820 变成肯定，从而处理进行到步骤 S821，在此确定是否已进行任何输入操作，以下载音乐段数据。在判断结果为肯定时，使一个对由输入操作指定的音乐段数据进行装入的命令，在步骤 S822 传送到主计算机 10。其后，该处理在步骤 S823 把标志 TERM11 设置成“1”以后，终止步骤 S80 的网络处理。

如果在步骤 S821 回答为否定，则处理分支到步骤 S824，在此确定是否已进行任何输入操作，以下载波形数据或参数数据。在判断结果为否定时，处理分支到步骤 S826，在此进一步确定是否已进行任何下载程序数据的输入操作。如果在步骤 S826 判断为否定，则处理分支到步骤 S828，以便执行一个相应于任何其他输入操作的处理。如果经步骤 S824

确定，已进行任何一个下载波形数据或参数数据的输入操作，就在步骤 S825，把一个对由输入操作指定的波形数据或参数数据进行下载的命令，传送到主计算机 10。如果经步骤 S826 确定，已进行任何一个下载程序数据的输入操作，就在步骤 S827，把一个对由输入操作指定的乐音发生程序或表演处理程序进行下载的命令，传送到主计算机 10。在这些操作之后，该处理在把标志 TERM11 在步骤 S823 设置成“1”以后，终止步骤 S80 的网络处理。

当在步骤 S820，由于把标志 TERM11 设置成“1”而使判断结果成为否定以后，处理分支到步骤 S829，以执行一个所下载数据的接收处理。此后，处理进到步骤 S830，在此确定是否已完成下载数据的接收。如果在步骤 S830 回答为肯定，则处理转到步骤 S831，以通知用户：已完成接收。其后，在步骤 S832 使标志 TERM11 复位成“0”，并使标志 TERM2 设置成“1”，然后该处理终止当前执行的网络处理。如果经步骤 S830 确定，尚未完成下载数据的接收该处理就终止网络处理，而不进行步骤 S831 和 S832 的操作。

当在步骤 S809，由于标志 TERM2 被设置成“1”而使判断结果为否定以后，处理就分支到步骤 S810，以检查标志 TERM3 的当前状态。如果在步骤 S810，由于标志 TERM3 设置成“0”而得出肯定的判断结果，则处理移到步骤 S840（图 13），以检查标志 TERM21 的当前状态。如果标志 TERM21 当前为“0”，则处理转到步骤 S841，以确定是否所接收的数据是音乐段数据。如果所接收的数据不是音乐段数据，则处理跳到步骤 S848，使标志 TERM1 和 TERM11 复位成“0”，和使标志 TERM3 设置成“1”，然后终止当前执行的网络处理。

如果经步骤 S841 确定，所接收的数据是音乐的数据，则处理转到步骤 S842，在此确定是否当前下载处理，是采用乐音发生器程序伴随方式。在回答为否定时，象不同于音乐段数据的其他数据被装入的情况一样，在进行步骤 S848 的操作以后，该处理终止步骤 S80 的网络处理。

如果经步骤 S842 确定，当前下载过程是采用乐音发生器程序伴随方式，则处理进到步骤 S843，以执行一个乐音发生器程序（TONE PGM）指定处理。这一指定处理，根据所接收音乐段数据的音乐段相关

数据 ( MUSIC MISC DATA ) 中所含的相应乐音发生器 ( 程序 ) 信息 ( TG INFO ), 参照用户概貌信息 ( USER PROFILE ), 识别一个最佳乐音发生器程序的名称和版本号, 以便把该最佳乐音发生器程序指定为下载的候选者。

从那些与 CPU、个人计算机和操作系统相适应的程序中, 选择一个最佳的乐音发生器程序, 它还满足下列条件:

( 1 ) 不需要大于用户存储容量的容量, 或需要诸程序中的最小存储量;

( 2 ) 能够发生那些满足用户期望乐音质量的乐音;

( 3 ) 如果存在多个满足上述条件的乐音发生器程序, 则应是其中的最新版本。

等等。

在以这种方式识别最佳乐音发生器程序的名称与版本以后, 处理进到步骤 S844, 在此检查一个装入终端设备的软件表 ( 尤其是, 在图 5C 所示的用户概貌信息的目录信息中的软件表 ) 或一个并入终端设备中的媒体目录表, 以确定是否所选择或所指定的最佳乐音发生器程序未储存在终端设备中。如果乐音发生器程序未储存在终端设备中, 则处理进到步骤 S845, 以发送一个指定所选最佳乐音发生器程序的下载命令。其后, 该处理在下一个步骤 S846 把标志 TERM2 设置成 “1” 以后, 终止当前执行的网络处理。如果相反, 软件表或目录表表明, 所选最佳乐音发生器程序储存在终端设备中, 则处理转到步骤 S847, 以便通知用户。其后, 在下一个步骤 S848 把标志 TERM1 和 TERM11 重置成 “0” 和把标志 TERM3 设置成 “1” 以后, 该处理终止当前执行的网络处理。

当在步骤 S840, 由于把标志 TERM21 设置成 “1” 而使判断结果成为否定以后, 处理就分支到步骤 S849, 以执行一个下载数据接收处理。此后, 处理进到步骤 S850, 在此确定是否已完成下载数据的接收。如果在步骤 S850 回答为肯定, 则处理转到步骤 S851, 以通知用户: 已完成接收。在下一步骤 S852, 使目录信息中的软件表 ( SOFT LIST ) 在步骤 S852 被更新, 以登记下载的乐音发生器程序 ( TONE PGM )。其后, 在下一个步骤 S853 把标志 TERM11 复位成 “0” 和把标志 TERM3 设

置成“1”以后，该处理终止当前执行的网络处理。

以上述方式实现乐音发生器程序的下载。然而，在下述场合可以略去这样的乐音发生器程序的下载：用户系统已在其中安装该最佳乐音发生器程序，或通常等效于或好于该最佳乐音发生器程序的类似程序或新版程序；或者安装一个具有功能上等效于或好于这个程序的程序的硬件乐音发生器。只有在详细地进行了上述环境检查以后，才能够进行一种是否应实现相应于系统状态的下载的确定的，或一种要下载的程序或数据的选择。此外，在主计算机中存有相同名称的不同版本的程序数据的情况，可以自动地选择最后的或最新的版本。还有，可以把指定要用的乐音发生器程序的数据，插入所装入音乐段数据的表演数据中，或表演部分的数据块中，以便通过检查指定的数据来选择用于下载的乐音发生器程序。此外，还可以引入一种所谓的“自动驾驶”功能，使处理按预定的顺序自动地进行。

当标志 TERM2 和 TERM3 都处于“1”时，则步骤 S810 的判断结果为否定，从而处理移到步骤 S860，在此确定标志 TERM22 当前是否为“0”。如果在步骤 S860 回答为肯定，则在步骤 S861 进一步确定是否当前的处理是采用波形数据伴随方式。如果是，则处理进到步骤 S862，以执行一个波形（WAVE DATA）指定处理。这一指定处理，根据所接收音乐段数据的音乐段相关数据（MUSIC MISC DATA）中所含的音色表，参照用户概貌信息（USER PROFILE），识别最佳波形数据的名称。按照通常与上述的关于最佳乐音发生器程序条件相同的条件，选择最佳波形数据。

在以这种方式识别了最佳波形数据的名称以后，处理进行到步骤 S863，在此确定是否最佳波形数据未寄存在波形表的目录中，或终端设备的记录媒体中。如果该波形数据未储存在终端设备中，就在步骤 S863 得出一个肯定的确定，然后处理进到步骤 S864，以传送一个指定最佳波形数据的下装命令。其后，在下一个步骤 S865 把标志 TERM22 设置成“1”以后，该处理终止当前执行的步骤 S80 的网络处理。

如果相反，波形表或媒体目录表明，经步骤 S863 确定，最佳波形数据储存在终端设备中，则处理转到步骤 S866，以便通知用户。其后，

在下一个步骤 S867 把标志 TERM2、TERM3、TERM11 和 TERM21 复位成“0”以后，该处理终止当前执行的网络处理。

如果经步骤 S861 确定，当前的处理不是采用波形数据伴随方式，则处理跳到步骤 S867，以便把标志 TERM2、TERM3、TERM11 和 TERM22 复位成“0”，并且终止当前执行的网络处理。

当在步骤 S860，由于标志 TERM22 被设置成“1”而使判断结果为否定以后，处理就分支到步骤 S868，以执行一个所装入数据的接收处理。此后，处理进到步骤 S869，在此确定是否已完成下载数据的接收。如果在步骤 S869 判断结果为肯定，则处理转到步骤 S870，以通知用户：已完成接收。在下一个步骤 S871，目录中的波形表（WAVE LIST）在步骤 S871 被更新，以寄存所接收的波形数据。其后，在下一个步骤 S872 使标志 TERM2、TERM3、TERM11 和 TERM21 复位成“0”以后，该处理终止当前执行步骤 S80 的网络处理。如果经步骤 S869 确定，尚未完成下载数据的接收，该处理就终止当前执行的网络处理，而不进行步骤 S870、S871 和 S872 的操作。

#### （应用处理）

图 15 是一个在每个终端设备中在步骤 S70 执行的应用处理流程图，其中，在步骤 S71 先执行音乐段数据表演处理，然后在步骤 S76 执行乐音发生处理，并在步骤 S79 执行其他的处理。在步骤 S79 的其他处理中，根据终端设备的类型和技术要求，进行不同于音乐段数据表演和乐音发生处理的其他处理；如果终端设备是一个博弈机，则可进行包括博弈本身处理在内的其他处理。

#### （音乐段数据表演处理）

图 16 是一个音乐段数据表演处理的流程图，其中先在步骤 S711 执行一个操作检测处理，以确定已由用户进行过何种操作。其后，在步骤 S712 确定，是否当前正在进行任何表演。如果在步骤 S712 确定，当前未在进行表演，则处理进到步骤 S713，以执行一个表演格式/环境设置处理，它设置各种表演条件，例如表演的拍子、音量和模式。此后，在步骤 S73 执行一个音乐段数据选择处理，以选择和装入音乐段数据。在下一个步骤 S715 执行一个演出开始操作处理，然后终止这个音乐段数据表

演处理。

如果在步骤 S712 判断结果为肯定，就在步骤 S716 执行一种表演事件发生处理，在此对在步骤 S73 的音乐段数据选择处理中装入的音乐段数据，相继地进行读出，以便根据读出的数据，以预定的计时产生一个表演事件，例如 MIDI 事件。根据所发生的表演事件，以后描述的步骤 S76 的乐音发生处理将控制图 3 的乐音发生器 113，以产生一个乐音。在步骤 S716 的表演事件发生处理以后，在步骤 S717 执行一个表演控制操作处理，以进行相应于 STOP 或 PAUSE 之类的用户输入的操作。此后，终止这个音乐段数据的表演处理。

下面参照图 17，描述步骤 S73 的音乐段数据选择处理的细节。首先，在步骤 S731，显示一个音乐段表，并且该处理等候，直至用户从所显示的表中选定一个音乐段为止。在步骤 S732 确定，用户是否已选定任何音乐段。如果尚未选定音乐段，就终止这一音乐段数据选择处理，而不执行其他的操作。如果经步骤 S732 确定，已选定任一音乐段，就读入所选择或所指定的音乐段的数据（MUSIC DATA）。其后，在步骤 S734，该处理对于在所读入音乐段数据中所含的版本号数据（VER NO ID 或 SONG VER NO ID）和乐音发生器信息（TG INFO），与软件表（SOFT LIST）上显现的乐音发生器程序（TONE PGM）的版本等之间的一致性，进行检查。然后在步骤 S735，确定是否在读入的音乐段数据与当前运行的乐音发生器程序之间的一致性不良。

如果由于在版本号或类似数据上出现不良的一致性而在步骤 S735 得出一个肯定的确定，则处理进到步骤 S736，在此装入和启动软件表（SOFT LIST）上的相应乐音发生器程序。这时，根据当前存储装载或用户概貌内容，选择是应装入该程序，以添加到当前运行的乐音发生器程序上，还是取代后者。在步骤 S736 以后，该处理移到步骤 S737。如果由于呈现良好的一致性而在步骤 S735 得出否定的确定，则处理直接转到步骤 S737，而不进行步骤 S736 的操作。

在步骤 S737，查明是否在表演事件数据或音色表（TONE COLOR LIST）中的音色指定数据的内容与此已装入的波形数据（WAVE DATA）之间，存在良好的一致性。其后，在步骤 S738 确定，是否存在

任何未装入的波形数据。如果在步骤 S738 回答为否定，则处理转到步骤 S741，以便进行音乐段选择终止处理，于是结束这种音乐段数据选择处理。如果在步骤 S738 回答为肯定，则处理转到下一步骤 S739，以便确定是否未装入的波形数据在波形表（WAVE LIST）上。如果是，则处理进到步骤 S740，以读入未装入的波形数据，然后执行步骤 S741 的音乐段选择终止处理。

如果经步骤 S739 确定，未装入的波形数据未在波形表（WAVE LIST）中，就在步骤 S742 装入替代的波形数据。例如，在可选择 5 个不同的钢琴音色 PIANO1 至 PIANO5 的场合，并且如果上述音乐段使用 PIANO3，而只有 PIANO1 装入终端设备中，就装入 PIANO1，以用于替换 PIANO3。其后，在步骤 S743，把替代波形数据的装入通知用户以后，处理转到步骤 S741 的音乐段选择终止处理，以终止该处理，并且显示读入的波形数据。

#### （乐音发生处理）

步骤 S76 的乐音发生处理用于发生一个乐音，其方法是：通过步骤 S716 的上述表演事件发生处理，根据以预定的时间发生的象一个 MIDI 事件之类的表演事件，控制乐音发生器。在此就一个只用编码器/检测器（CODEC）的 D/A 转换器作乐音发生器 113 的由软件运算地生成波形数据的实例，描述乐音发生处理。如果一个共鸣板或外部乐音发生器被作为乐音发生器 113 而连接，则用普通的驱动器软件执行乐音发生处理。

在由软件运算地生成波形数据的场合，在 RAM 103 中提供各种存储区，用作 16 种音色数据 PD1 至 PD16 和 n 种波形数据 WD1 至 WDn 的寄存器，32 个通道 CH1 至 CH32 的通道寄存器，和多个输出缓冲器。

每个音色数据 PD1 - PD16 都包括指定所给音调(pitch)范围的波形的数据（音调范围波形指定数据）；对用于给出颤音效果或类似效果的低频振荡器（LFO）进行控制的数据（LFO 控制数据 OD）；用于控制滤波器包络的生成以时变一种音色滤波特征的数据（FEG 控制数据 OD）；用于控制音量包络的生成以控制音量变化特征的数据（AEG 控制数据 OD）；用于按照检测触键速度来改变乐音或类似音的上升速度的接触控制数据（接触控制数据 OD）；和包括波形样品形成分辨控制

数据在内的其他数据(其他数据 OD)。附在每个上述数据尾端的“OD”代表在受到象调制之类的控制以前的“原始数据”。在本实施例中,按照响应一个乐音发生指令而生成的接触数据、音调数据等,通过处理这些原始数据,形成用于乐音发生器的乐音形成数据。

此外,按照输入的乐音音调数据,由音色数据 PD1 - PD16 之一的音调范围波形指定数据,指定波形数据 WD1 - WDn 中的任何一个。

每个通道寄存区都存储多个乐音发生控制数据,它们包括音符数,波形指定数据, LFO 控制数据,滤波包络控制数据(FEG 控制数据),振幅包络控制数据(AEG 控制数据),音通(note-on)数据和其他数据,以及一个由 CPU 101 在执行程序期间使用的工作区。在通道寄存区中的波形指定数据、LFO 控制数据、FEG 控制数据和 AEG 控制数据,都是通过处理上述的各个原始数据而得到的乐音形成数据,此后将在每个这类数据的尾端附上字母“D”来指明。

随后,用输出缓冲区作乐音波形形成输出缓冲器 X。如下面所述,每当通过算术运算而形成用于各个乐音发生通道 SD1、SD2、SD3、... 的乐音波形样品数据时,这些数据都是逐个通道地累加的,并且所得的累加数据都存入输出缓冲器中。指定两个或多个这类输出缓冲器作乐音波形形成输出缓冲器 X,并把它们用于波形计算操作。按照最简单的形式,可以用两个输出缓冲器 X 来提供一种双缓冲区结构,以便在其中一个缓冲器 X 存储的数据被一个波形重放段重放的同时,把通过波形形成操作得到的新波形样品数据存入其中另一个缓冲器 X。

输出缓冲器可以具有任选的大小,例如 100 字, 500 字, 1K 字或 5K 字。本专业技术人员熟知:较大的输出缓冲器在发出一个乐音时会导致较大的时间延迟,而较小的输出缓冲器在临时增加要作的计算量时,会导致减少的时间余量和差的响应。然而在象本发明一样的情况下,在此执行以定序程序为基础的演出或自动的演出,而不需要实时处理能力,从而输出缓冲器可以作成大的,因为能够通过使演出时间提前而有效地调节乐音发生的延迟。

另一方面,在乐音发生程序用于需要实时演出能力的键盘演出或类似演出的场合,为了防止乐音发生延迟, 100 至 200 字词范围的输出缓



冲器大小是适宜的。这样的输出缓冲器大小适合于再现 40 至 50KHz 的取样频率，而对较低的取样频率，则需要小得多的输出缓冲器。

在用上述各种数据执行步骤 S76 的乐音发生处理方面，终端设备的 CPU101 对每个乐音发生通道进行操作，以便按照预定的时间点集体地在运算上产生或形成多个（例如 100 个）乐音波形样品。也就是说，对每个乐音发生通道，都同时发生相当于 100 个 D/A 转换器取样周期的乐音波形样品。

这样，在每个预定的计算点，都对全部通道进行乐音发生处理。在所发生的乐音波形样品数被设置成 100 的场合，对这些样品依次累加，作为 D/A 转换器的 100 个取周期的通道特有累加值，并且存入上述的输出缓冲器 X 中。在为全部乐音发生通道完成样品积加以后，储存在输出缓冲器中的乐音波形样品，由波形再现部分按照一个取样周期一个样品的方式进行读出，并且通过 D/A 转换器把它们供给用于可听声再现或发声的音响系统。控制本实施例中预定的计算点，使之按下述的时间间隔发声：当用输出缓冲器依次地进行乐音发生处理时，能够完整地读出和再现全部乐音波形样品，而不会中途丢下或断开。

在上述的安排下，只有在集体地发生多个乐音波形样品的情况下，才需要对每个通道进行预备性操作，从而有可能减小花在预备性操作上的算术运算或计算的时间对整个计算时间的比例，从而显著减小开销。这就能够显著改进所产生的乐音波形样品的质量，和增加可同时产生的乐音数。

此外，每当收到输入的数据（在这种情况下是 MIDI 事件数据）时，都只对最多到时间点的样品进行波形算术运算或计算。在预定的计算点，只对那些预定的多个乐音波形样品（即，相当于每个输出缓冲器 X 大小的样品）中的未计算过的样品，进行乐音波形样品计算。

在乐音发生处理中，要对每个当前正在发声的乐音发生通道进行波形样品计算，存在涉及随输入数据而变的特殊形式乐音发生（音调偏移或音量变化）的键通或键断事件的各个通道，要比不随输入数据而变化继续进行乐音发生的其他通道，需要更多的算术运算。在这种情况下，就把计算点设置成以固定的时间间隔发生，由那些涉及乐音发生的形式

变化的通道占据大部分的操作周期，这对那些能够在有限时间内完成所需算术运算的乐音发生通道，可最终减少其数目。尤其是，对各个要开始进行乐音发生的通道来说，需要进行许多初始化操作，例如初始化地址计数器和包络发生器和发生“F”数等，从而需要长的操作时间。因此，通过每当输入 MIDI 事件数据时，都集体地进行乐音波形计算，就能有效地避免乐音发生通道数的减少。

也就是说，当输入音通事件数据时，就把音通事件的音符数 NN 和速度 VEL 存入各个寄存器中，并且还把该事件的发生时间作为时间 TM 存入相应的寄存器中。其后，根据所寄存的音符数 NN 进行乐音发生通道赋值，并且把这样赋值的乐音发生通道 (CH) 的通道号作为“CH<sub>i</sub>”而存入寄存器中。其次，对通道 CH<sub>i</sub>，在通道寄存器中设置相应于音符数 NN 和速度 VEL 的乐音控制数据。这样设置的乐音控制数据是，按照音通数 NN 和速度 VEL 的数值，通过对那些相应于已接收音通事件的 MIDI 通道的预先存储的音色数据的数据，进行处理而得到的乐音形成数据 D。乐音形成数据中的波形指定数据 D，按照在相应于音符数 NN 的乐音发生中使用的一个波形，规定波形数据 WD<sub>1</sub> 至 WD<sub>n</sub> 中的一个数据。

在设置了乐音控制数据以后，就设置一个用于通道 CH<sub>i</sub> 的音通标志，然后通过运算形成一个乐音波形。根据一个要写入当前可用输出缓冲器 X 的完整波形的一部分，来实现这种运算形成，这个一部分是尚未计算的，相当于事件发生时间 TM 以前的周期的；并且把这样计算的部分波形写入输出缓冲器中。部分波形是一个在检测新接收数据时，其数据已被固定，从而成为可计算的波形部分。此外，要通过运算形成的部分波形是，一个直到所接收音通事件的发生时间 TM 为止的波形部分。

当输入一个音断事件时，把音断事件的音符数 NN、速度 VEL 和发生时间 TM 存入各个寄存器中。其后，对发出音符数 NN 的一个乐音发生通道 (CH) 进行识别，并且把所识别的通道 CH<sub>i</sub> 号存入相应的寄存器中。此后，复位用于所识别通道 CH<sub>i</sub> 的音通标志，并执行乐音波形计算。在这种波形计算中，对相当于在时间 TM 以前一段时期的未计算的波形部分 (部分波形) 进行计算，并且用与上述方式类似的方式把它写

入输出缓冲器 X 中。此外，还对通道 CHi 执行一个预定的释放处理。

图 20 是一个说明上述乐音发生处理的时序图，其中，作为双缓冲区结构提供的输出缓冲器分别用图中 (e) 部分的 A 和 B 表示的。从缓冲器 A 和 B 再现所需的时间是分别用 TA、TB 表示的，并且在本实施例中时间 TA 等于 TB ( $TA = TB$ )。

首先，如图 20 的 (a) 部分所示，在分配给第一输出缓冲器 A 的从点 t0 到点 t1 的计算时间范围内的点 ta，用一个 MIDI 接收部分收到两个 MIDI 事件以后，一个乐音发生器 (T.G.) 驱动器部分进行如 (b) 部分所示的处理，并且一个乐音发生器 (T.G.) 部分，对如 (c) 部分所示的相当于从点 t0 到点 ta 的时期接收的输入的乐音波形样品 A1，进行计算。

乐音发生器驱动器部分接收来自输入缓冲器或来自个人计算机键盘的数据，把所接收的数据分配到乐音发生器通道之一中，并且按照输入的数据执行一个从音频参数到乐音发生器参数的转换。此外，乐音发生器部分接收该乐音发生器参数，并处理波形数据，以形成要真正听到的乐音波形样品。一个 LPF 部分把混入的噪音分量从所形成的乐音波形样品中清除，并且把来自 LPF 部分的输出数据写入第一和第二输出缓冲器 A 和 B 中。上述的乐音发生器驱动器、乐音发生器和 LPF 部分，是一些由执行应用程序的 CPU101 实现的函数。

其后，当在点 tb 收到一个 MIDI 事件以后，乐音发生器驱动器部分进行类似于上述处理的处理，并且乐音发生器部分对相当于从点 ta 到点 tb 期间接收的输入的乐音波形样品 A2，进行计算。其后，在到达点 t1 时，乐音发生器部分对相当于从点 tb 到点 t1 期间接收的输入的乐音波形样品 A3，进行计算。在这种情况下，当在点 ta 和 tb 收到键通事件时，就在这个 t1 点进行乐音发生初始化和运算操作。此后，LPF 进行一个滤波操作，从而对第一输出缓冲器 A 完成所需的乐音波形样品形成。

其后，如图 20 的 (a) 部分所示，在一个分配给第二输出缓冲器 B 的从点 t1 到点 t2 的计算周期中，在点 tc 收到三个新的 MIDI 事件。因为乐音发生器部分当前正在计算乐音波形样品 A3，故暂时在输入缓冲器中保存输入的事件数据，直到把计算时间分配给该数据为止。在完成由乐

音发生器部分进行的乐音波形样品 A3 的计算和由 LPF 部分进行的滤波操作时, 就由乐音发生器驱动器部分从要处理的输入缓冲器中读出该事件数据, 并且乐音发生器部分, 对相当于从点 t1 到点 tc 期间接收的输入的乐音波形样品 B1, 进行计算。即使在计算操作中发生某些时间延迟, 也不影响乐音发生时间, 因为还在输入缓冲器中写入了输入事件的发生时间。

此后, 在计算乐音波形样品 B1 期间接收 4 个新的 MIDI 事件, 并且处理这些事件数据, 以便在计算乐音波形样品 B1 以后, 与上述情况类似地计算相应的乐音波形样品。

按照相同的方式, 形成乐音波形样品 B2, 用作相当于在从点 tc 到点 td 期间接收的输入的样品; 形成乐音波形样品 B3, 用作相当于在从点 td 到点 te 期间接收的输入的样品, 并且形成乐音波形样品 B4, 用作相当于在从点 te 到点 t2 期间接收的输入的样品。然后, 形成乐音波形样品 A5, 用作相当于在从点 t2 到点 t3 期间接收的输入的样品。

正如从上面描述看到的那样, 每当输入数据时, 该处理都在一段先于输入数据发生的时间计算波形样品, 从而能够用分散的方法进行波形样品的计算。这样, 以预定的时间间隔在每个计算周期进行的操作量是受到适当限制的, 这样, 即使发生许多输入数据, 例如涉及乐音发生形式变化的键通事件数据时, 也有可能避免麻烦, 例如可同时发生乐音数的减少。

为了在以预定的时间间隔产生诸计算点, 并且在每个计算周期以算术运算集体地形成预定数目的诸乐音波形样品的情况下, 相继地产生乐音, 就必须在完成以前的乐音波形样品的形成之前, 提供预定数目的诸乐音波形样品。然而, 如果乐音发生通道的数目比较多, 并且因此而使波形计算操作量过大, 就会出现一个问题: 不能及时提供全部通道的乐音波形样品, 使所发生的乐音产生一个不合乎需要的中断或断开。

为了避免上述的麻烦, 就把本发明的处理设计成确定: 是否能在 D/A 转换器的数据转换时及时地提供预定数目的乐音波形样品。如果确定, 在数据转换时能够及时地提供这些乐音波形样品, 就选择其指定乐音在当前时间正在发生的全部乐音中是重要性比较小的, 一个或多个乐音发

生通道。然后针对指定的通道，该处理即刻计算那些相当于波形初始期间的衰减波形样品。

因为只对所指定的乐音发生通道计算短期衰减的波形样品，故能减少花在这个通道上的所需计算时间，从而可在 D/A 转换器的数据转换时基本上及时地提供预定数目的乐音波形样品。

一般说来，在处理中作为“重要的”而确定的乐音有：

- (1) 当前有大音量的乐音；
- (2) 其增高部分刚开始要发声的乐音；
- (3) 当两个或多个部分的乐音正在发声时的最低音调乐音（低频乐音）；
- (4) 当两个或多个部分的乐音正在发声时的最高音调乐音（主要乐音）；和
- (5) 当两个或多个部分的乐音正在发声时的独奏部分的乐音。

可以在波形样品计算以前，按照其中发生的乐音的重要性递减次序，配置诸乐音发生通道，于是从最重要的乐音到最不重要的乐音相继地实现样品的计算。如果不能及时地完成全部通道的波形样品计算，就中止计算，以便只用迄今已形成的乐音波形样品来发生乐音。这样，假如需要中断样品的计算，就只有那种重要性比较小，从而影响比较小的一个或几个乐音被减弱或压抑。

为了在以预定的时间间隔发生诸计算点，并且通过算术运算集体地形成预定数目的乐音波形样品的情况下，相继地发生乐音，就必须在完成以前的乐音波形样品的形成之前，提供预定数目的乐音波形样品。然而，如果乐音发生通道的数目比较多，从而波形计算操作量过大，或者如果由于在不是乐音发生处理（例如定序器处理）的其他处理中花费太多的时间，而不能及时提供预定数目的乐音波形样品，就甚至在正在处理的过程中，也会读出一个或多个乐音波形样品，从而导致不想要的噪音。

这个问题能够解决，方法是由 CPU101 指令波形再现部分，发送存储于任何一个输出寄存器中的数据。在这种情况下，能够在用作一个读出部分的，或用作一个要从其中重复地读出数据的重复读出部分的再现

部分中，设置那些本身用于储存预定数目的所形成乐音波形样品的输出缓冲器 X 中的地址。此外，可以用这样的方式作出一个读出部分的预约：能够在一个当前正在从其中读出数据的特定读出部分之后，立即读出在这些地址的数据。

在上述配置下，读出部分预约容许所形成的乐音波形样品寄存在用于预约的输出缓冲器中，并且在当前正在读出的特定波形之后立即读出。如果对预定数目的乐音波形样品的计算尚未及时完成，则这样一种预约寄存无效，从而有可能防止发生噪声，任何一个仍然处于处理过程中的乐音波形样品的发声都会按另一种方式产生这种噪声。这会暂时中断一个乐音的发声，但在取样频率例如是 44.1KHz 的情况下，通过把中断时间限制到只有几个取样周期，会明显降低该中断引起的不良影响。如上所述，通过限制那些发生乐音的通道数，可以实现中断时间的限制。在完成了计算以后，进行预约寄存，使相应的波形样品发声。

如上所述，为了在以预定的时间间隔发生诸计算点，并且以算术运算集体地形成预定数目的乐音波形样品的情况下，或者在根据每个输入数据的发生来执行波形样品的计算的情况下，相继地发生诸乐音，就必须在完成以前的乐音波形样品的形成之前，提供预定数目的乐音波形样品。

根据以前的乐音波形样品的形成被结束时的一个时间点（以前的乐音波形样品的结束时间点），指定一个象用于一个特定乐音波形样品的计算点之类的点，该点早于一个按照完成一个波形形成所需的时间来规定的结束时间点。因为通过 CPU 101 检查再现部分的当前状态（标志），识别结束时间点，以便借此检验什么时候样品再现时间段已移到下一个时间段；所以实际的结束时间点会延迟，直至 CPU 检测到再现部分的状态（标志）的变化为止。此外，因为这种时间延迟取决于 CPU 101 实现上述检测的时间，故各个形成的样品会呈现随各个检测时间而异的不均匀延迟。

如果根据上述不均匀延迟的结束时间点发生诸计算点，就不能得到正确的计算点。尤其是，如果根据明显延迟的时间来发生诸计算点，则在计算开始与波形数据供应之间的总时间长度会明显缩短，从而可同时

发生的乐音数会暂时减少。

为了解决这一问题，CPU 101 就在以前检测到再现部分的状态变化时，在存储器中预先存储多个时间点，并且均分所存储的时间点，从而 CPU101 可预测下一次检测到这样的一种变化时的时间点。因为预测的时间点相当于根据再现部分中所规定结束时间点的时间延迟的平均点，故能够检测一个在预测时间点以前是一预定时间的点，用它作一个实际上正确的结束时间点。这样，根据正确的结束时间点，产生计算点，用这种方法，均分所检测的结束时间点，以减少它的不均匀性，使分配给各个计算点的操作周期有效地均匀化，于是以稳定化的方式进行乐音形成操作。

可以不根据每个输入数据的发生，为每个时间部分集体地进行波形样品的计算。在这种情况下，可以优先地发生一个触发信号，以便按照输入数据的数量来提出计算点。在另一项修正中，在诸预定的计算点之间的每个时间部分可以分成“ $n$ ”个相等的部分，以使用相当于“ $n$ ”个部分的时间间隔来进行波形形成计算处理，并且可对已在上一次或第“ $n$ ”次计算中完全形成的一个时间部分的乐音波形样品的发声，进行预约寄存。

下面参照图 18 和 19 的流程图，详述乐音发生处理的根本操作。

首先，在步骤 S761 检查再现部分的再现状态。如果再现部分已推进到下一个要再现的部分，则处理转到下一个步骤 S762；如果否，则处理跳到步骤 S763。

波形再现部分，从一个由 CPU101 指定的 RAM103 的特定区域，以预定的取样频率，逐一地读出波形样品，并且把读出的样品供给用于音响再现的 D/A 转换器。当从指定的区域读出波形样品时，波形再现部分就从 CPU 101 指定的要再现的另一区域接收一个预约。在完成了前一个区域的再现以后，由再现部分同样地逐一读出这样预约的下一个区域的波形样品，并且把它们供给用于音响再现的 D/A 转换器。该再现部分超前于下一个要再现的部分在此意味着，在完成上一次作为要再现部分而指定的特定区域的再现以后，该再现部分已移到另一个作为下一个要再现的部分而预约的特定区域。在这种情况下，能够一次预约多个特定的

RAM 区域，并且按照预约它们的顺序再现之。

其后，在步骤 S762，根据当前的和以前的超前检测的时间，预测要在下一次检测的这样一种超前的时间（下一次超前检测时间），并且指定一个在预计检测时间以前是一个预定时间的时间点，把它用作下一个计算点。可以预测下一个超前检测时间，其方法例如有：根据多个包括当前时间和上一个时间在内的超前检测时间，通过“最小二乘法”，计算一个具有小误差的近似值；或用一个二次函数或类似函数去逼近检测时间上的变化。在发生与超前检测之间的一段时期中，由于各个操作步骤的位置与状态上的差异，而在波形再现部分发生不均匀的时间延迟，使超前检测时间包含不规则的不均匀性。因此，逼近函数的计算涉及一种平均该不均匀检测时间的运算。

上述的在预测检测时间以前的“预定时间”是一个为形成一个乐音波形而分配的时间，并且取决于波形形成的必需算术运算量，这是由要发生的乐音期望数、算术运算的期望质量等确定的。“预定时间”的长度可以通过用户在键盘上的选择而加以固定，或可变地设置；也可由 CPU101 考虑用两个或多个同时运行的程序来自动地设置。

其后，在步骤 S763，在下一个计算点与由一个计时器指示的当前时间之间作一个比较，以确定是否当前时间已到达下一个计算点。如果判断结果是肯定的，则该处理进行步骤 S764 至 S769 的操作。首先，在步骤 S764，决定特定的计算次序，其中，当前正在发声的通道应经历波形样品计算，以便过后为每个当前发声的通道而形成多个样品的波形数据。

这就是说，要用下述方式设置计算次序：从一个在音乐上重要的乐音开始，或者从一个若在那一时刻消失或压抑则会呈现重要音乐麻烦的乐音开始，相继地进行波形样品计算。然后，在步骤 S765，确定是否能够在预定的计算时间（即在步骤 S762 描述的预定时间）范围内，完成全部当前发声通道的波形样品计算。如果以否定来应答，则把一个或多个要在过后或晚于任何其他通道的时间进行波形样品计算的乐音发生通道，指定为其乐音要消失的通道（乐音消失通道），以便按下述方式减少算术运算量：能够在预定的计算时间范围内完成全部当前发声通道



的波形样品计算。

在步骤 S78，实际进行波形样品计算。尤其是，对未计算过的波形样品进行计算，并且把它们写入当前可用的输出缓冲器 X 中，使缓冲器 X 装满所计算的波形数据，从而为乐音发生作好准备。这种操作相当于形成波形 A3 或类似波形，如上面关于图 20 的所述。

然后在步骤 S767，使每个写入装满输出缓冲器 X 的样品经历一种低通滤波（LPF）操作，以便从其中去除高频分量。此后，在步骤 S768，预约存储低通滤波波形的输出缓冲器 X，用作下次再现波形的一个特定存储区，以便在完成当前再现和业已预约的存储区中的波形再现以后，再现这一存储区。其后，在步骤 S769，除了迄今用作缓冲器 X 的缓冲器之外的另一个输出缓冲器，被清零和新置成输出缓冲器 X，以便为下一个要重现的部分而准备一个波形。

如果经步骤 S763 确定，当前的时间尚未到达计算点，则乐音发生的处理结束，而不进行其他的操作。

图 19 是一个在音通、音断和乐音发生器处理中进行的波形形成计算处理（步骤 S78）的流程图。对这种处理，预置一个波形计算时间范围，如上所述。即，在象音通处理之类的 MIDI 数据接收处理期间执行本程序流的情况，波形计算时间范围相当于上述的部分波形；而在乐音发生处理的步骤 S76 执行本程序流的情况，则波形计算时间范围相当于缓冲器 X 中一个未计算部分的波形样品。因为按照在以前执行乐音发生处理时确定的计算次序进行计算，故在接收 MIDI 数据时不为全部通道确定新的计算次序。每当一个新的音通事件发生并且被分配到一个特定的通道时，则该特定的通道被首先置于（给出一个第一顺序）计算次序中，并且各个其他通道的顺序往下移动一位。

首先，在步骤 S781，为一个首先置入计算次序的通道的第一波形样品，作计算准备。计算准备包括设置各种数据，例如上一次的读出地址、各种包络 EG 的数值和状态（增高、释放等）和处于可达状态的 LFO 值等，和把这些数据装入 CPU 101 的内部寄存器，供计算时直接使用。其后，在步骤 S782，对 LFO、滤波 G 和音量 EG 进行波形计算，以形成 LFO、FEG（滤波包络）和 AEG（振幅包络）波形的样品。把 LFO

波形添加到计算指定时间范围所需的“F”数、FEG波形和AEG波形上，以便调制各个数据。对每个在步骤S765被指定作乐音消失通道的乐音发生通道，按照一种在该时间范围内迅速衰减的音量EG，计算一种阻尼的AEG波形。

其后，在步骤S783，把F数重复地添加到用作初始值的上一次读出的地址上，以便为各个在该时间范围内的样品产生读出的地址。此外，根据读出地址的整数部分，从音色数据区中的波形存储区读出波形数据；并且根据读出地址的小数部分，在诸读出波形样品之间进行内插，以便形成该时间范围内的全部内插样品。例如，如果计时器范围相当于花在100个样品上的时间，则通过这一步骤的操作而集体地形成100个样品。因为在该时间范围内的多个样品的操作中，按照完整的系列操作来进行相应于读出地址的F数添加和内插，故只需要一次性地把读出的地址读入CPU寄存器中，从而总的说来，能够明显地提高处理的速度。

此后，在步骤S784，使在该时间范围内内插的样品经受一种音色滤波操作，在此根据上述的FEG波形进行样品的音色控制。在下一个步骤S785，对滤波后的样品进行一种振幅控制操作，以便根据上述的AEG（振幅包络）和音量数据来控制样品的振幅；然后进行一种累积写入操作，以便把得出的振幅受控样品添加到在指定通道的输出缓冲器X中存储的相应样品中。因为在这一处理中，振幅的控制和对输出缓冲器X中相应样品的添加都是连续地进行的，故有可能使装入CPU寄存器的必需样品数减至最少，从而明显提高处理速度。

进行步骤S783至S785的样品形成操作，可基本上形成预定时间范围内的全部样品，而那些由于步骤S782的音量EG波形计算而具有充分低的AEG波形电平，从而具有充分低的音量的样品，是不作进一步计算的，从而能够减少必需的运算量。特别是，在乐音发生通道已根据步骤S765的指定而发生一种阻尼AEG波形的场合，常常可以在预定的时间范围内于中途得到充分的衰减。

其后，在步骤S786，确定是否要继续进行波形形成计算处理时把波形在该时间极限范围内供给波形再现部分，和是否应中断波形样品计算。在该时间极限范围内及时地提供波形在此意味着：从特定的存

储区在当前再现以前形成波形样品的再现部分,能够在一个新的缓冲器 X 的区域准备一些随后的波形样品,并且在完成从特定存储区再现以前形成波形样品之前,预约用于随后再现的缓冲器 X 的新区域.如果在步骤 S786 确定,不会及时提供波形,并且不应进一步继续计算,就在步骤 S788 中断波形样品的计算,并且波形计算处理结束.

如果在步骤 S786 确定,可以继续计算,就在步骤 S787 进一步确定,是否已对全部的指定通道完成了波形样品的计算.如果回答为否定,就在步骤 S789 指定,按计算次序在下一次配置的乐音发生通道中的第一个波形样品(给出下一个计算顺序),并且准备好在这下一个通道中计算波形样品.在完成这些准备时,处理回到步骤 S782,为这个通道而重复步骤 S782 至 S785 的操作.用这种方法,对全部指定的乐音发生通道,重复地进行步骤 S782 至 S787 的操作,并且每当对这些通道中的一个通道进行这些操作时,都在步骤 S785 把在预定的时间范围内组合地形成的样品,累积地添加到储存在缓冲器 X 中的相应样品中.

当在步骤 S787 确定,已对全部的指定通道完成波形样品的计算时,这一波形计算处理就结束.这样,对全部指定的通道,现在新存储了相当于该时间范围的预定累计值数量的形成波形样品.

当根据步骤 S786 的肯定确定,中断波形样品计算时,就已对每个指定的通道,新储存了相当于该时间范围的预定累计值数量的形成波形样品.

对每个其计算顺序迟于计算中断时间的通道,不实现波形样品的形成,结果,已通过该通道发生的乐音会消失.然而,因为在步骤 S764 使这些发生较小重要性乐音的通道处于计算次序中的较迟顺序,故能使由中断产生的不利影响减至最小.在步骤 S788 以下述方式设置通道寄存器 CH:甚至在下次执行波形形成计算处理之中和之后,如此排斥在计算之外的通道仍然处于乐音消失状态.

用上述方式,能通过 CPU101 的操作而运算地形成乐音波形样品.

虽然在上面使用由终端设备处理的用户概貌信息,通过装入一个最佳乐音发生器程序或最佳波形数据,描述了本发明,但这样的用户概貌信息可以预先存储在主计算机 10 中,或者可以在 LOGIN 时间从终端设

备之一传送到主计算机 10，以便当有来自终端设备的装入请求时，主计算机 10 能够参照用户概貌信息去选择和传送最佳数据或程序。

在一个修正方案中，主机可以发送一个用于各个音乐段的可用波形数据表，以便终端设备请求只传送一个该设备不处理的选择部分的波形数据。

正如迄今已描述的那样，本发明容许装入最佳数据或程序，并且有利于根据用户系统中的系统和程序处理条件来选择要装入的数据。这就有可能防止没关系的数据被装入，还防止增加网络处理量。

此外，还有可能避免从 CD - ROM 之类的记录媒体装入不需要的数据或程序。例如，在采用一种 PCM 乐音发生器的场合，可以在各个终端设备中存储标准的波形数据；并且对一个使用特殊波形数据的音乐段，可以把特殊的波形数据与音乐段数据一起装入。这种配置可防止音乐表达由于在乐音发生器与 MIDI 数据之间的误匹配而变坏。

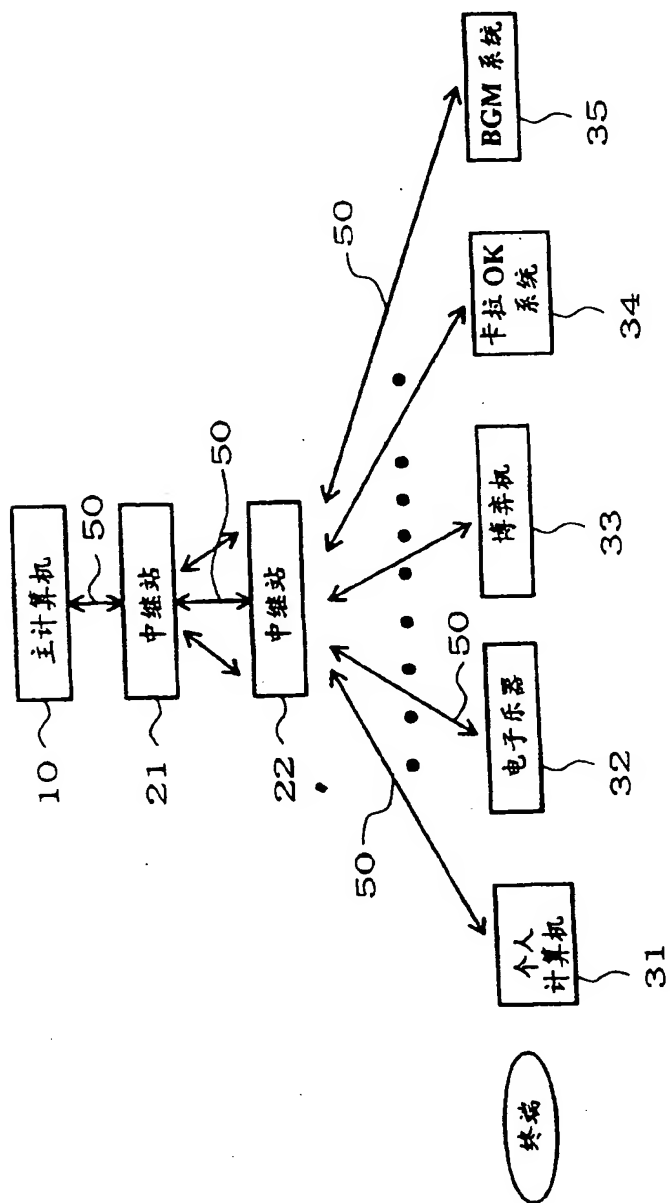


图1

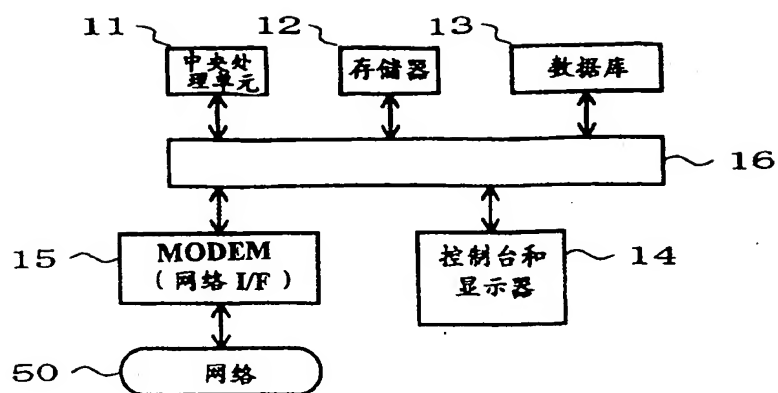


图 2A

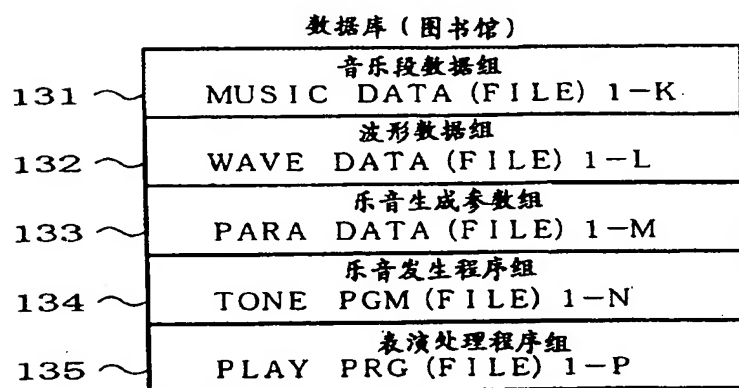


图 2B

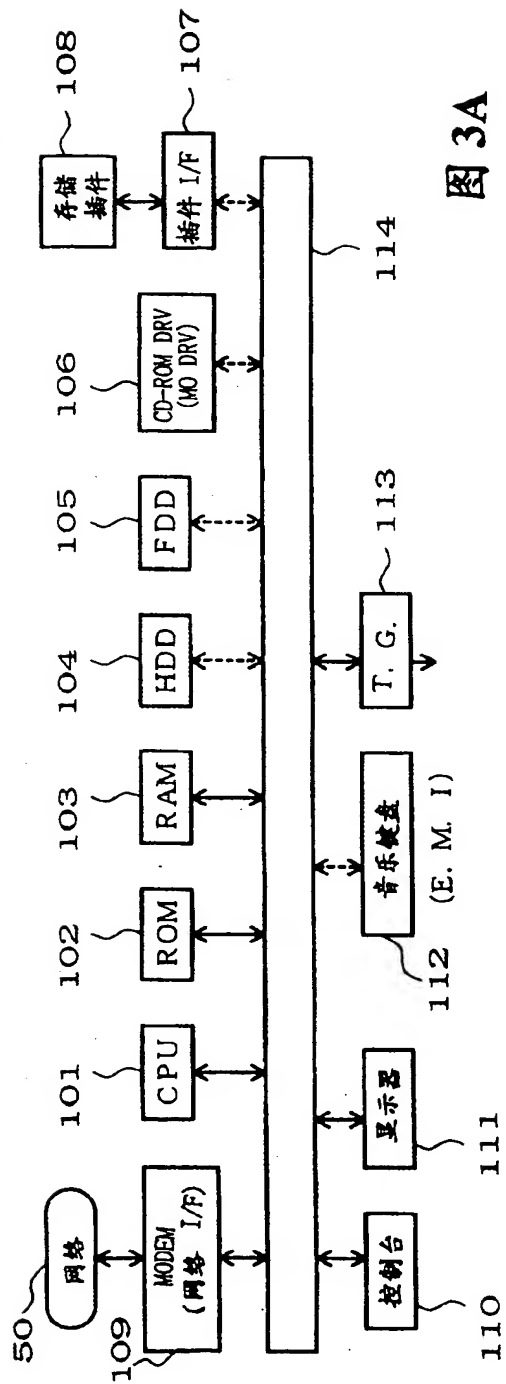


图 3A

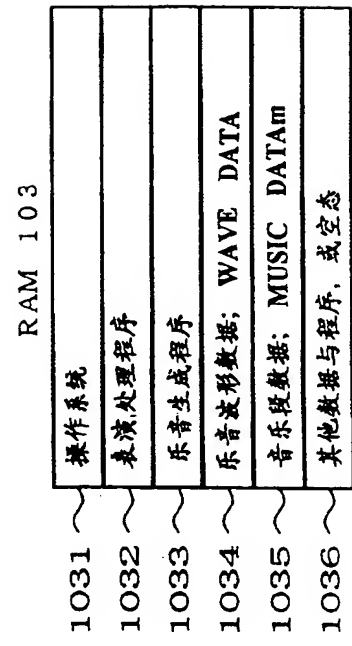


图 3B

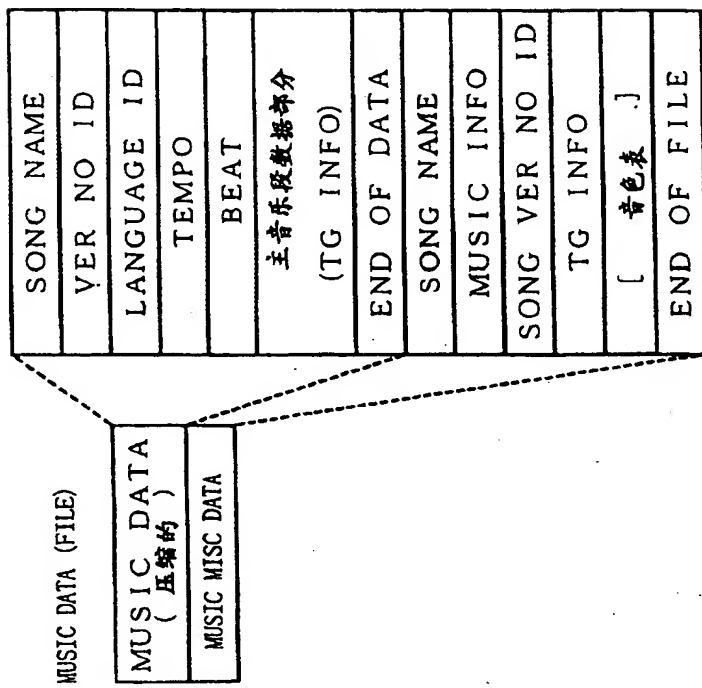


图 4A

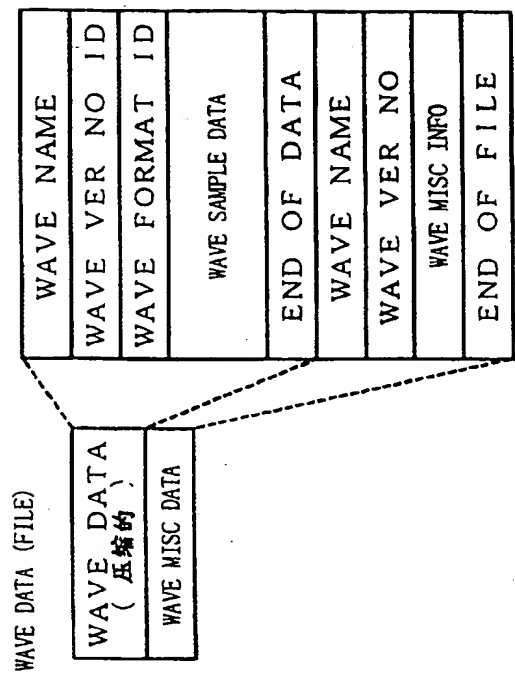


图 4B



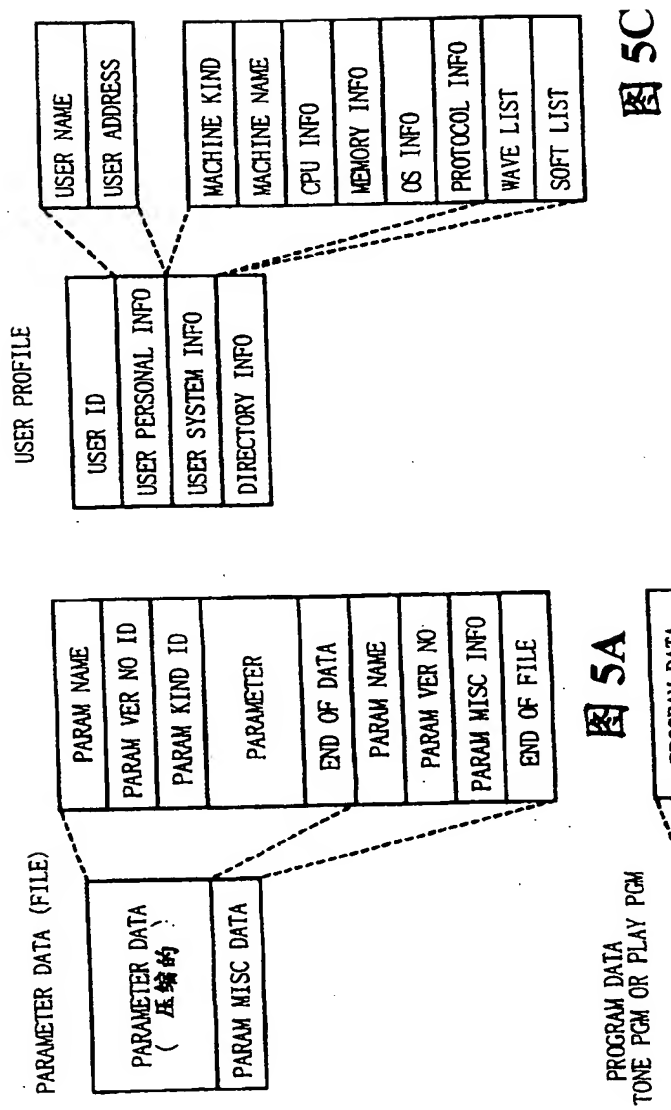


图 5A

图 5C

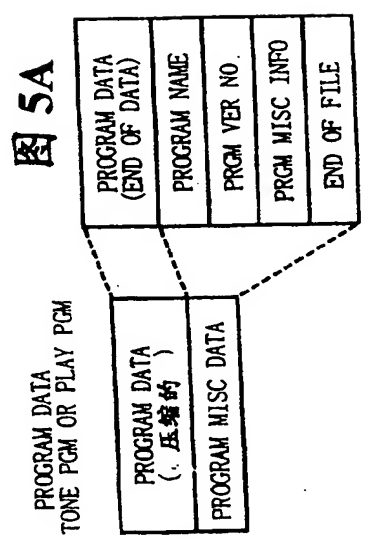


图 5B

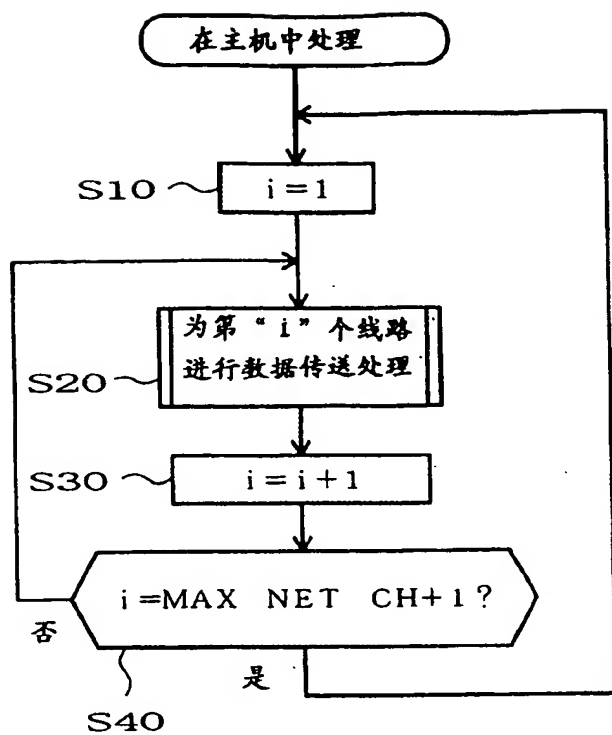


图 6

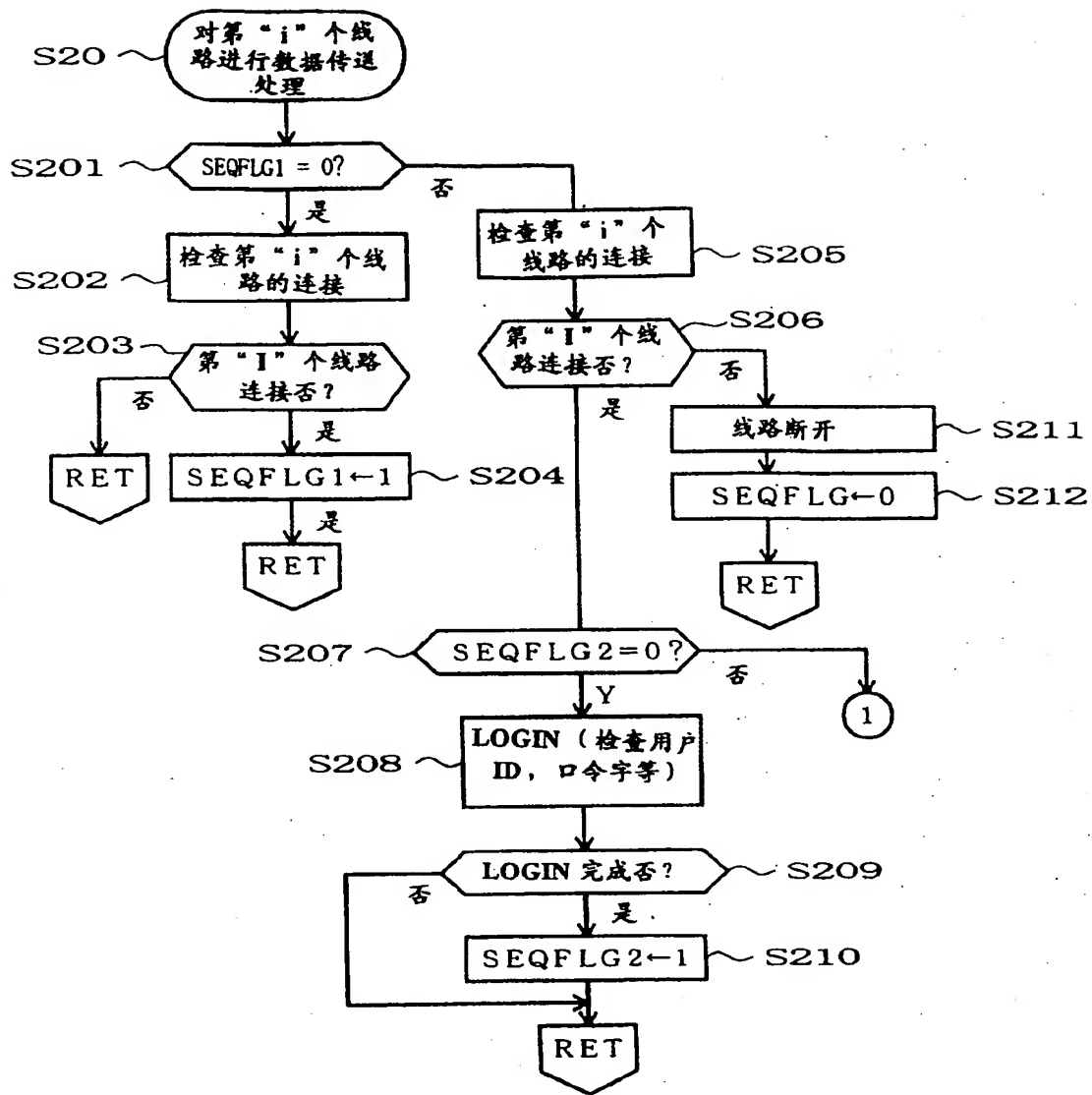


图 7



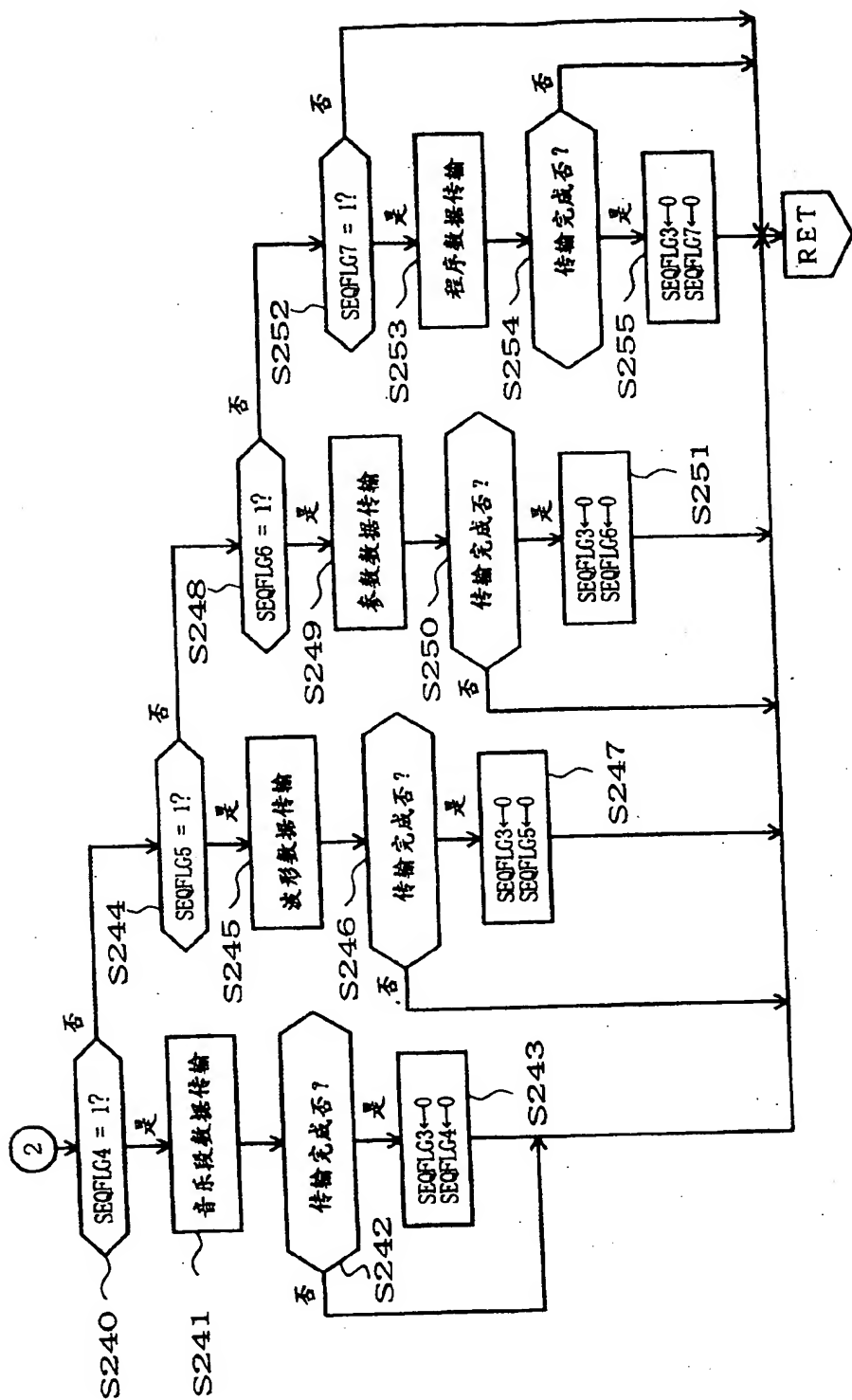


图 9

在终端设备中处理

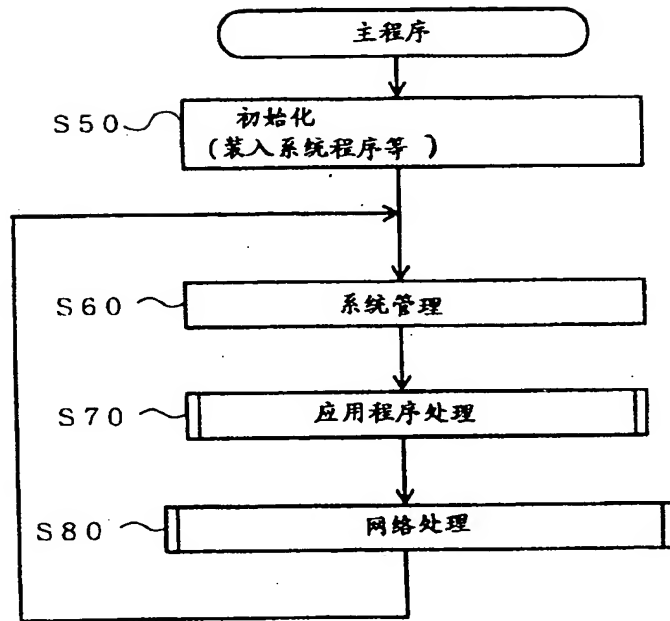


图 10

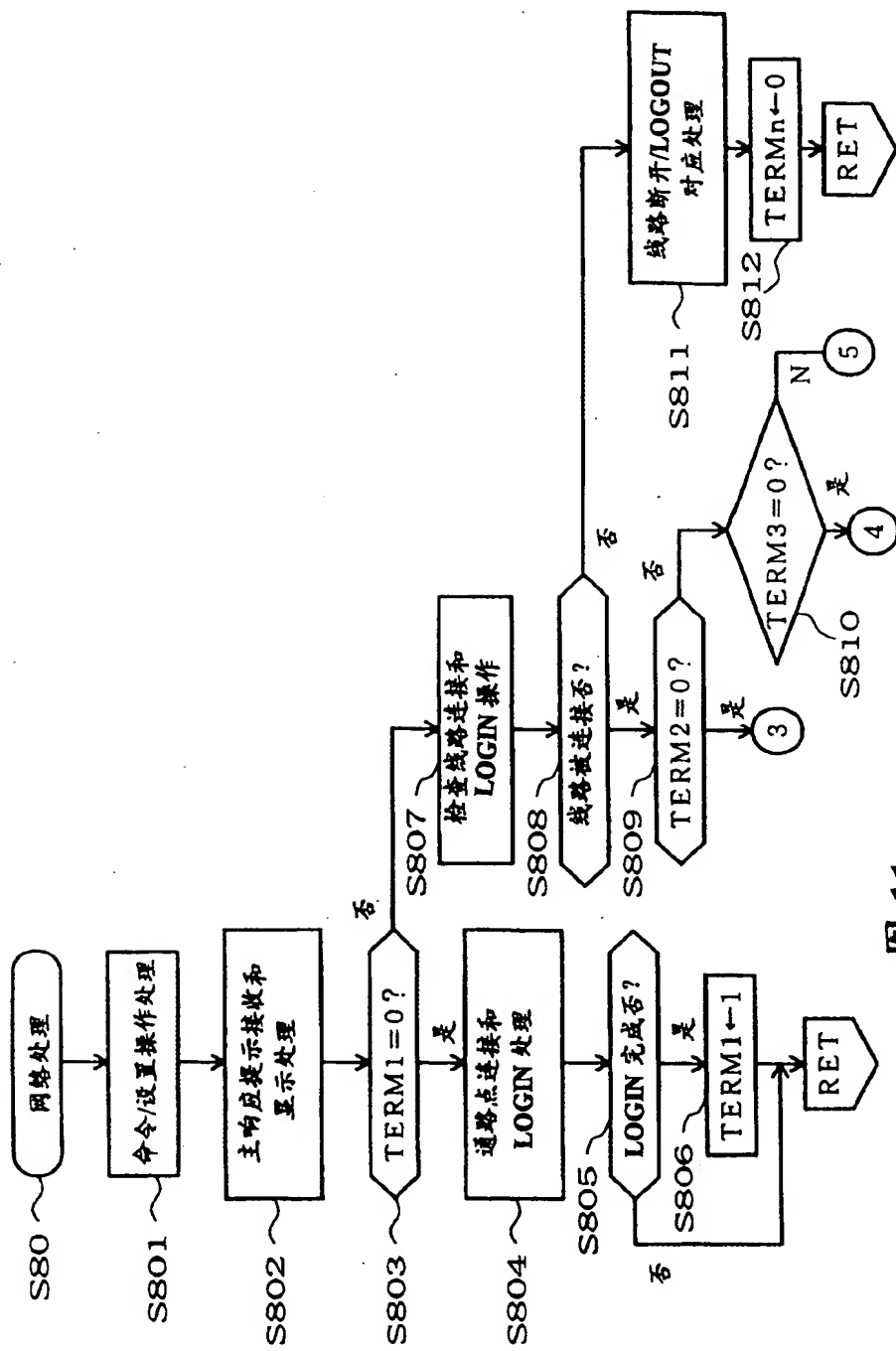


图 11

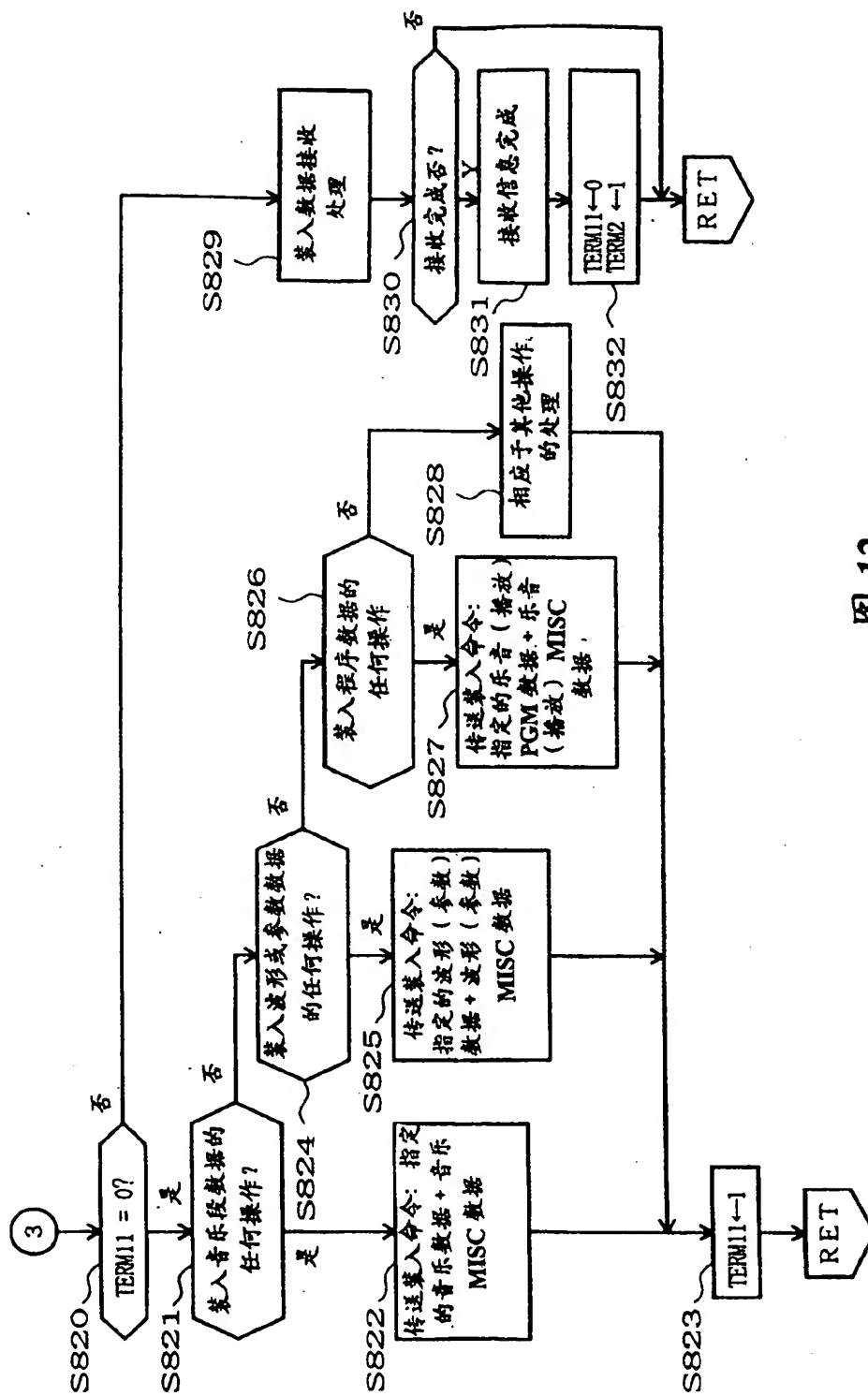


图 12



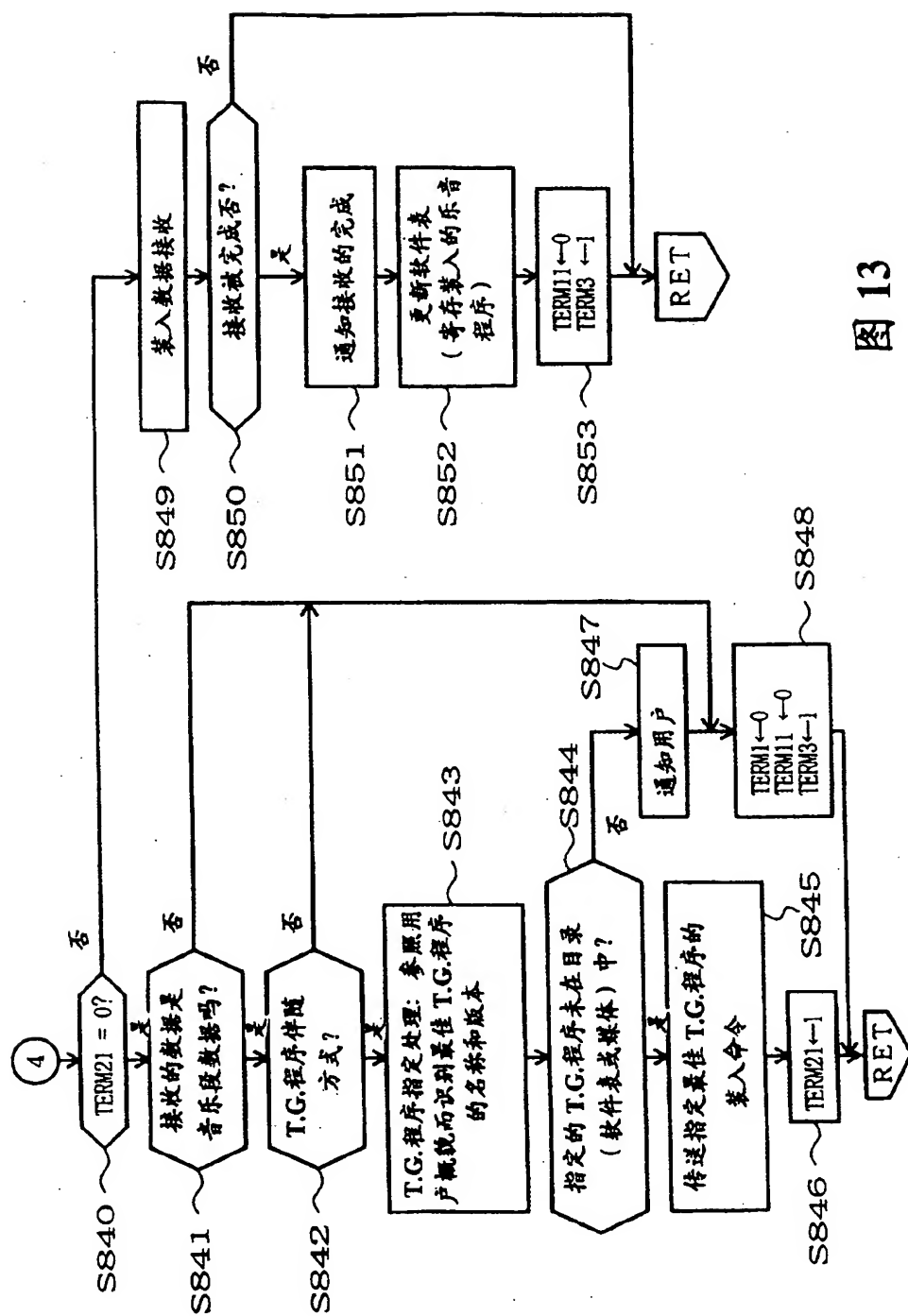


图 13

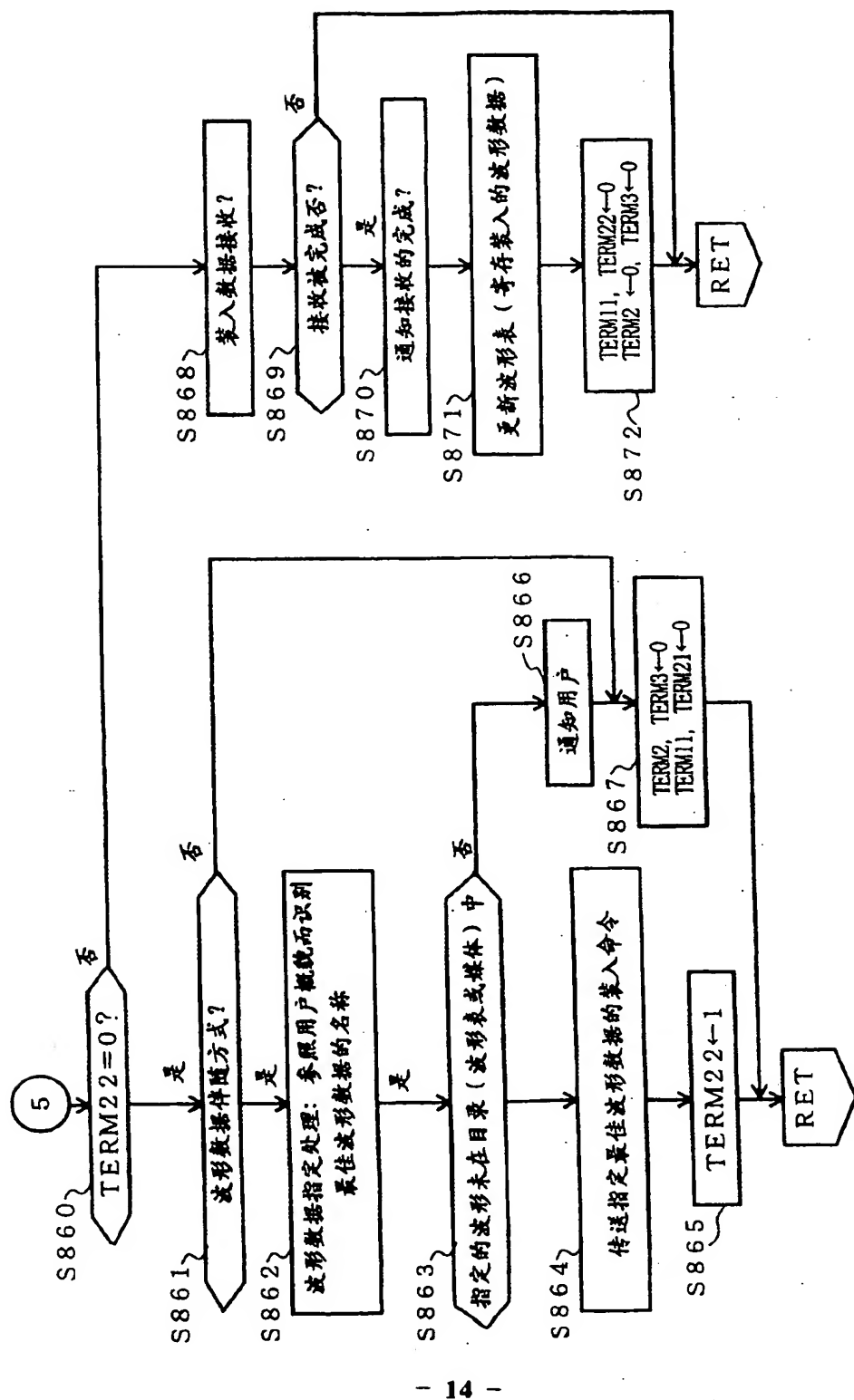


图 14

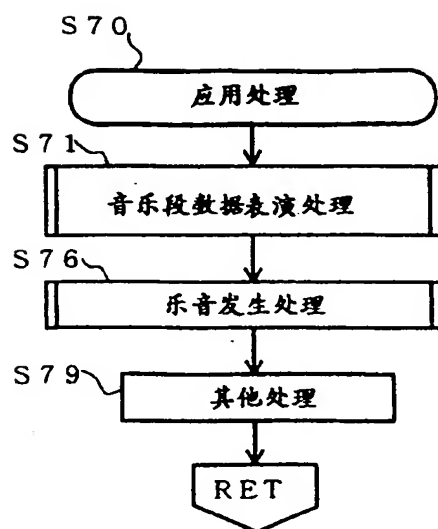


图 15

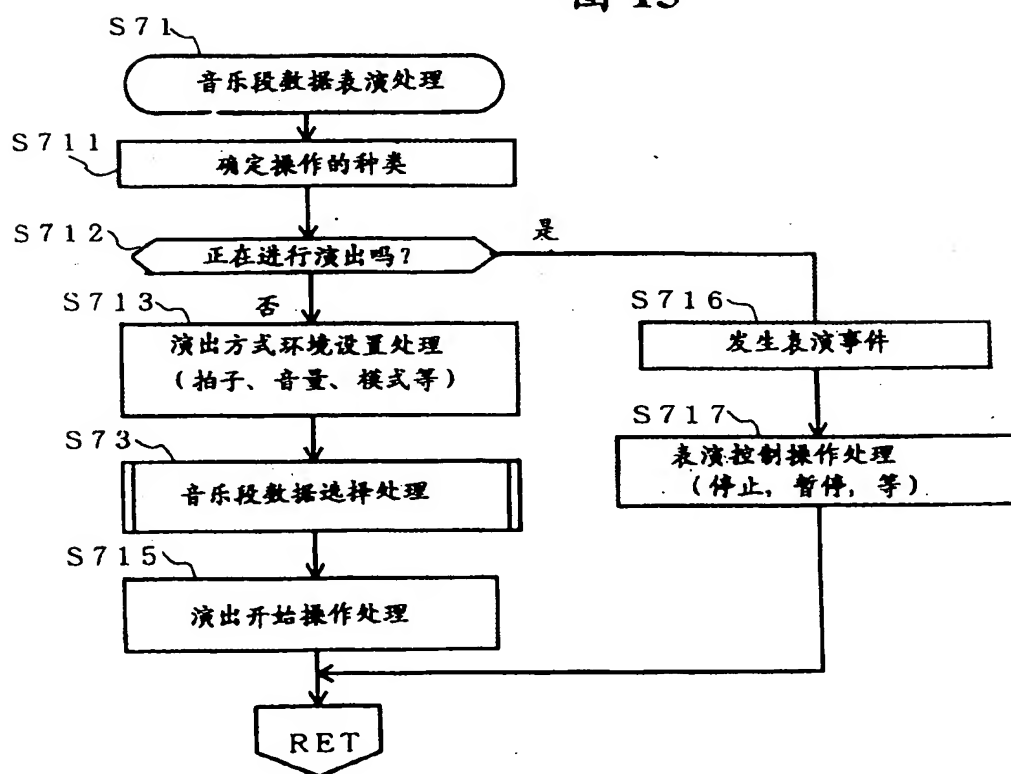


图 16

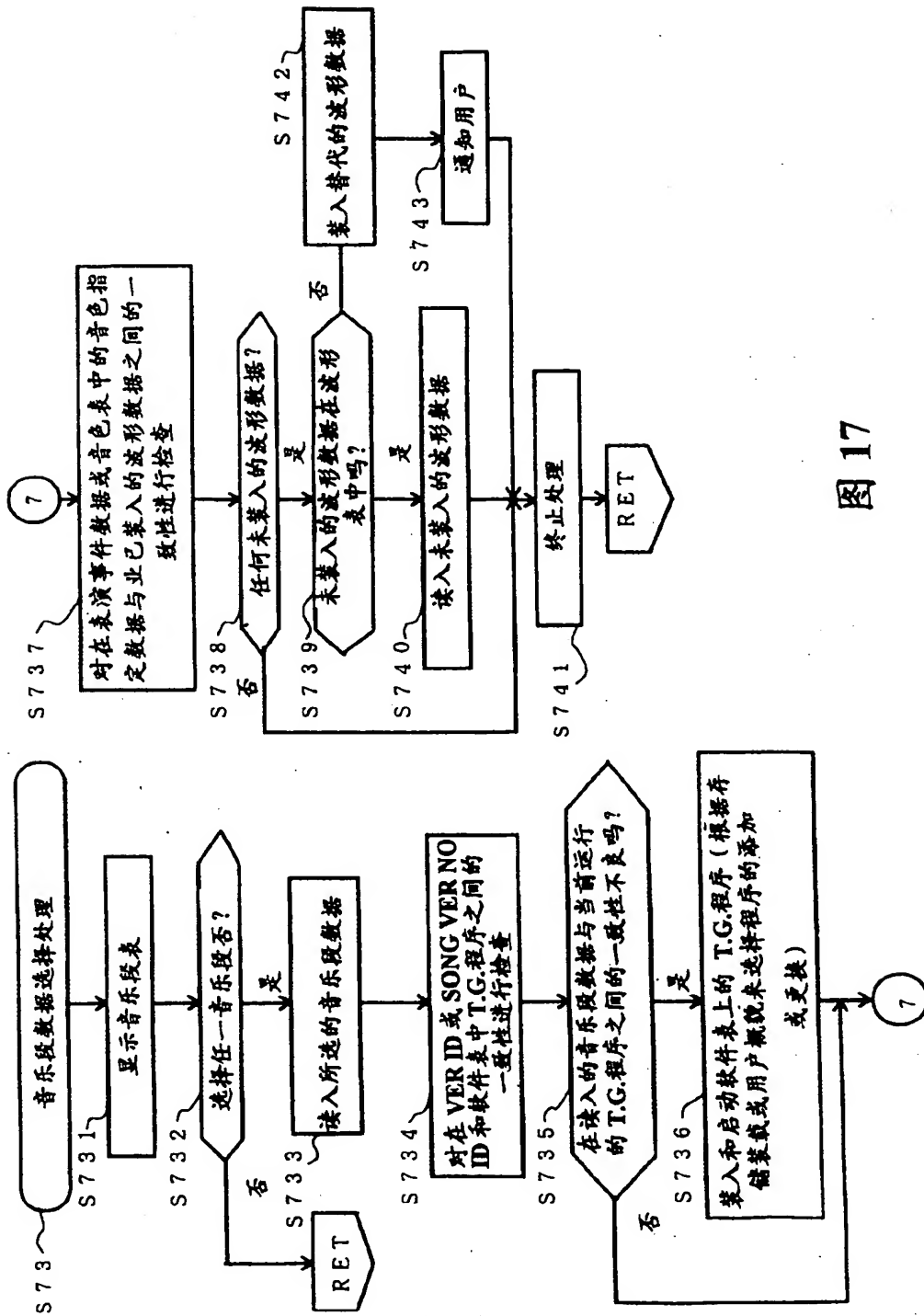


图 17

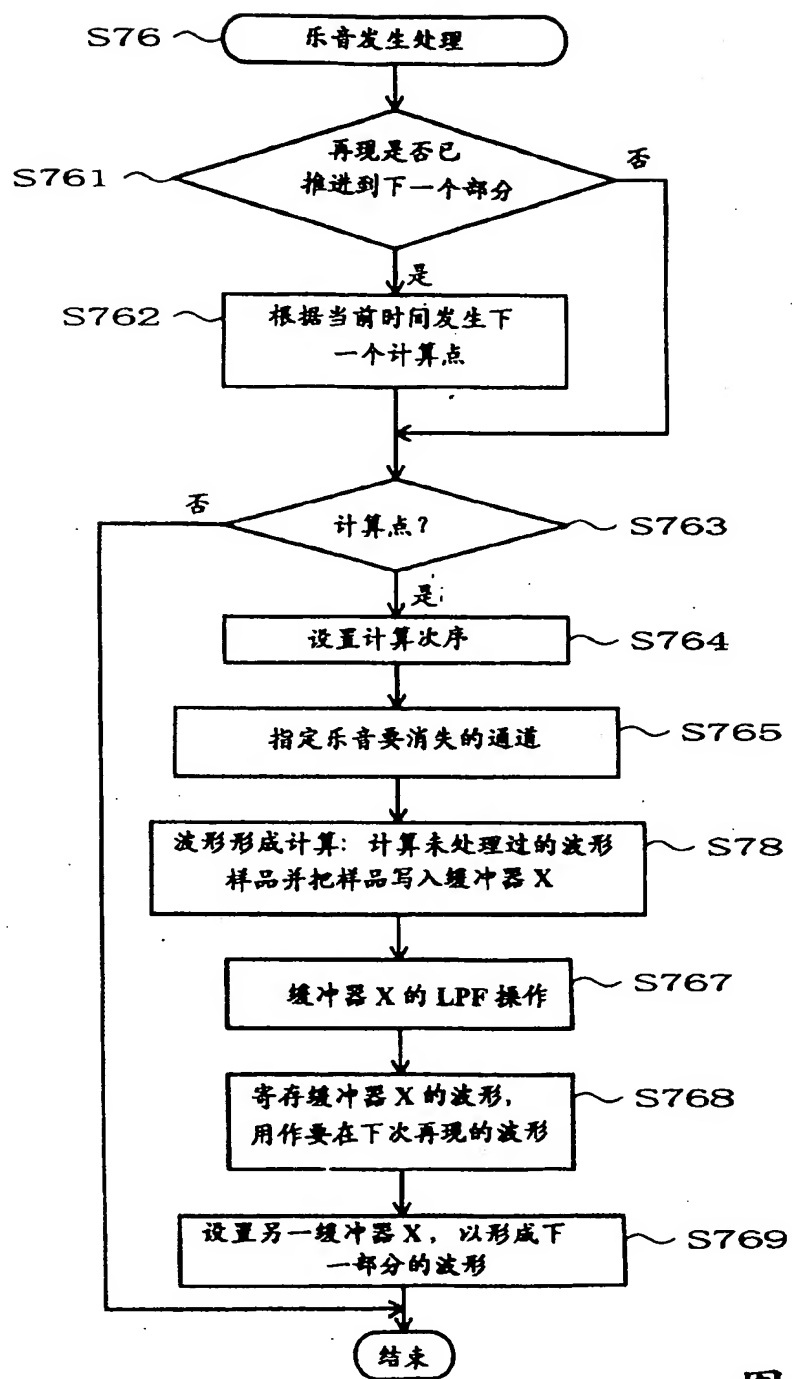


图 18

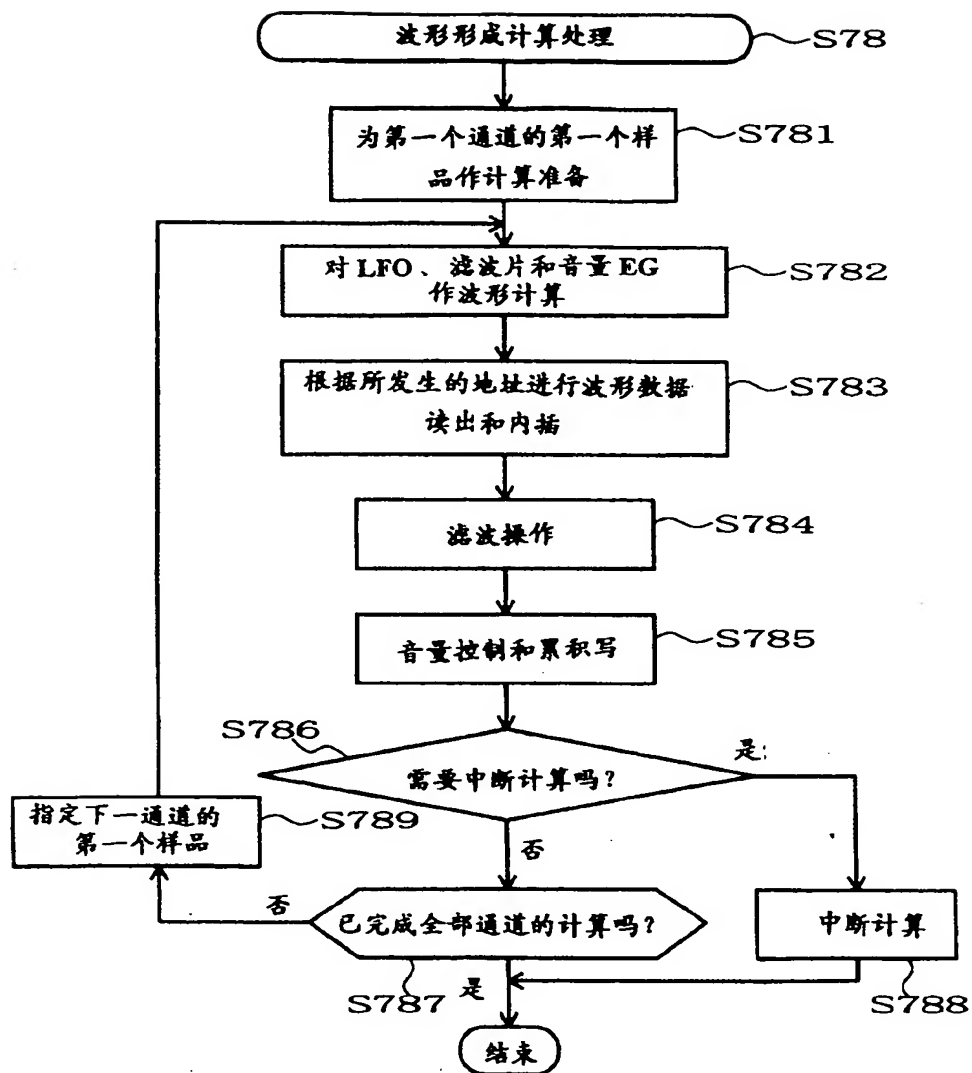


图 19

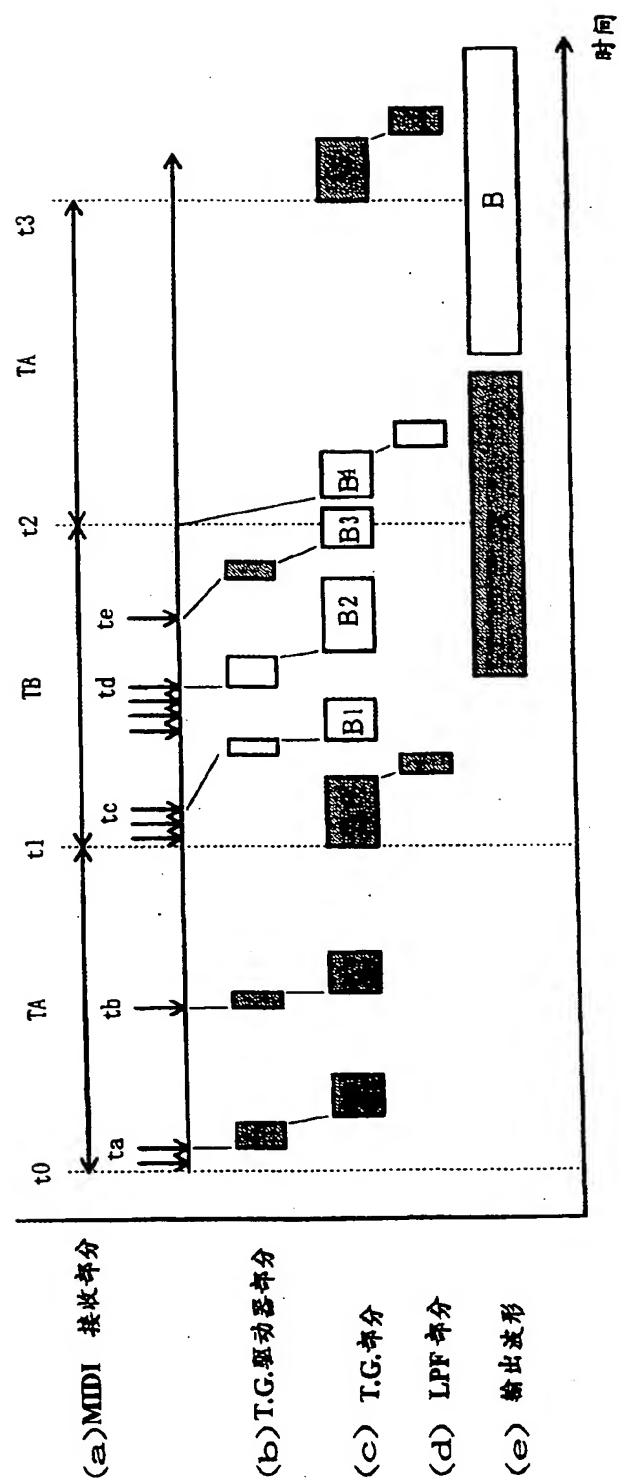


图 20